

COLEGIO DE EDUCACIÓN PROFESIONAL TÉCNICA DEL ESTADO DE MÉXICO.



PLANTEL. ING. BERNARDO QUINTANA ARRIOJA.

LA VIDA DEL VIDEO JUEGO.

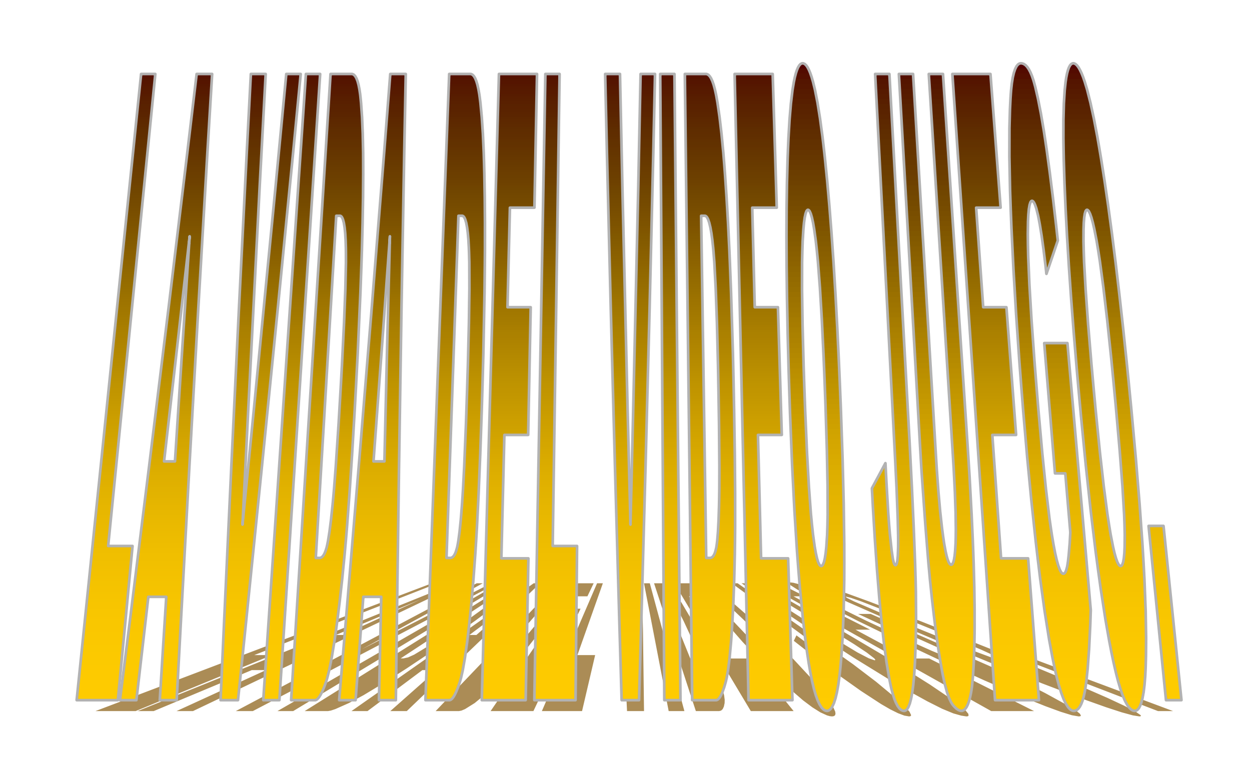
TRABAJO RECEPCIONAL PARA OBTENER EL TÍTULO DE PROFESIONAL TÉCNICO EN INFORMÁTICA.

P R E S E N T A :

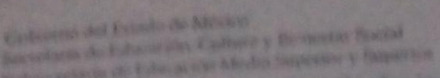
VALDÉS VARGAS AXEL ABRAHAM.

ASESOR: ALCÁNTARA SERRANO JUANA.

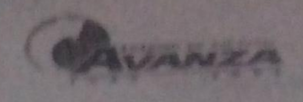
CUAUTITLÁN IZCALLI ESTADO DE MÉXICO JUNIO 2005.


LA VIDA DEL VIDEO JUEGO.









Colegio de Educación Profesional Técnica del Estado de México

Organismo Público Descentralizado del Gobierno del Estado de México
Plantel: Ing. Bernardo Quintana Arrioja No. 109

CARTA RESPONSIVA

Cuautitlán Izcalli, Estado de México a 17 de Junio del 2005

ARQ. CARLOS ABEL ORTEGA JUÁREZ
DIRECTOR DEL PLANTEL
PRESENTE

ATN.: LIC. ALICIA CAMARENA MAQUEDA
JEFE DE PROYECTO DE SERVICIOS EDUCATIVOS

Por este conducto me permito comunicar a usted, que revisé el trabajo recepcional con el tema:

"La vida del video juego".

que presenta (n) el pasante (s) Valdés Vargas Axel Abraham.
con número de matrícula 021090554, para obtener el Título de Profesional Técnico en:

Informática

y considero que dicho trabajo reúne satisfactoriamente todos los requisitos para ser discutido en el Examen Profesional correspondiente, por lo anterior le otorgo mi voto aprobatorio.

ASESOR DE TESIS

Lic. Juana Alcántara Serrano.

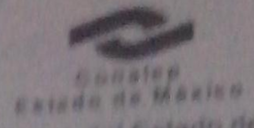
PROFR. DE SEMINARIO DE TITULACION

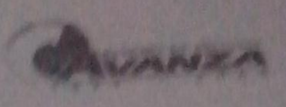
Marta Claudia Miranda Espinoza

COORDINADOR DE LA CARRERA

Ing. Ma. Lourdes Galvan González







Gobierno del Estado de México
Secretaría de Educación, Cultura y Bienestar Social
Subsecretaría de Educación Media Superior y Superior

Colegio de Educación Profesional Técnica del Estado de México

Organismo Público Descentralizado del Gobierno del Estado de México

Plantel: Ing. Bernardo Quintana Arrioja No. 109

AUTORIZACIÓN DE IMPRESIÓN

Cuautitlán Izcalli, Estado de México a 27 de Junio del 2005

LIC. ALICIA CAMARENA MAQUEDA
JEFE DE PROYECTO DE SERVICIOS EDUCATIVOS
P R E S E N T E:

Por este conducto notifico a usted, que el trabajo recepcional:

"La vida del video juego".

presentado por los (las) alumnos (as)

Valdés Vargas Axel Abraham.

de la carrera de Profesional Técnico en:

Informática.

ha sido revisado y cuenta con la autorización de la Coordinación a mi cargo, para imprimirse con fecha:

29.07.05

ATENTAMENTE

Dedicatoria...

A LA VIDA.

Quiero expresar un profundo reconocimiento a todas aquellas personas que voluntaria o involuntariamente, al paso del tiempo, han logrado que crezca como individuo, gracias a ello hoy he podido llegar a este momento en mi vida.

La vida sin inspiración no es nada,
pues cada segundo de ella es contundente,
ya que mis pasos son parte de la jornada,
que me hacen cada vez más fuerte.

Fuerza para seguir perseverante,
con caídas y tropiezos
ya que sin ellos no aprendería a ser constante,
seguir hoy y mañana para no quedar en pedazos.

Dulce seria la vida sin reto alguno,
pero es precisamente lo que le da emoción,
solo se vive una vez y no hay retorno,
por lo que cada segundo se acelera mi corazón.

Vida lléname de esperanza,
para poder lograr mis metas,
que es sufrir y regocijarme sin añoranza,
pues soy tú servidor fiel, que te necesita.



Agradecimientos.

A MIS PADRES:

Sabiendo que no existirá una forma de agradecer una vida de sacrificio y esfuerzo, quiero que sientan que el objetivo logrado también es de ustedes y que la fuerza que me ayudo a conseguirlo fue su apoyo.

Porque gracias a ese apoyo y consejo he llegado a realizar una de mis metas, la cual constituye la herencia más valiosa que pudiera recibir; con admiración y respeto:

Axel Abraham Valdés Vargas.

A MI ASESORA:

Como muestra de mi cariño y agradecimiento, por todo el apoyo brindado y porque hoy veo llegar a su fin una de mis metas de mi vida, le agradezco la orientación que me ha otorgado.

GRACIAS.

A MI HERMANO.

Como muestra de mi cariño y aprecio, por todos los momentos, en los que me apoyaste en la realización de esté trabajo, además como pauta para continuar está gran hermandad que se a forjado de ese ayuda mutua.

Mads.

ÍNDICE.

CAPÍTULO I. COMPUTACIÓN EN NUESTROS DIAS.

CAPÍTULO II. PROGRAMACIÓN Y SUS APLICACIONES EN LA ACTUALIDAD.

CAPÍTULO III. PROGRAMACIÓN APLICADA EN VIDEO JUEGOS.

CAPÍTULO IV. REALIZACIÓN DE UN VIDEO JUEGO.

PRÓLOGO.

La programación nace con la aparición de las computadoras, con un sistema de tarjetas perforadas las cuales transmitían lo que se deseaba realizar, después llego el lenguaje binario con la aparición de bulbos para mandar las señales a nuestros ordenadores, con el tiempo a evolucionado la codificación de sus lenguajes.

El tema es muy interesante, complejo, además se considera la madre de la informática dado que sin él no existiría ninguno de los programas que facilitan el trabajo en la oficina y en muchas otras áreas, donde en la actualidad se depende de una PC y su diverso software.

Posee grandes ventajas para realizar diversas estructuras con una función específica para la que se codificó como por ejemplo: tenemos procesadores de textos, presentadores gráficos, hojas de cálculo, reproductores de audio y video, etc.

Como gran desventaja, la programación es compleja y contiene exigencias que requieren de ciertos conocimientos en el área, de tal forma que se dificultará el desarrollo al no conocer palabras reservadas u otras reglas de sintaxis.

Este material proporciona como beneficios ampliar conocimientos, e inducir a las nuevas generaciones a buscar y aprender más sobre la programación.

Es un tema que más que novedoso, permite ver la evolución de los lenguajes ya que la tecnología siempre tendrá un avance continuo y existirá la necesidad de prosperar a la par.

INTRODUCCIÓN.

El tiempo ha demostrado solo una constante, la cual es una evolución en la tecnología, ya que somos testigos de grandes avances en los últimos años, de los que formamos parte, pues nos hemos convertido en partícipes de la automatización, de tal manera que nuestro entorno esta rodeado de ella, por lo tanto la mayoría de personas tiene un radio, un televisor, un horno de microondas, una computadora, etc. Algunos analistas mencionan que quien no posea una PC será considerado un analfabeta por no estar actualizado en el mundo moderno.

La computación, conjunto de conocimientos científicos y de técnicas que hacen posible el tratamiento automático de la información por medio de ordenadores. La informática combina los aspectos teóricos y prácticos de la ingeniería, electrónica, teoría de la información, matemáticas, lógica y comportamiento humano. Los aspectos de la informática cubren desde la programación y la arquitectura informática hasta la inteligencia artificial y la robótica.

La mayoría de áreas de trabajo necesitan un computador para realizar sus diversas funciones, por la necesidad de manejar una gran información en bases de datos, compartir investigaciones en las redes, manipular grandes máquinas para labores peligrosas que se le dificultarían al el ser humano, de esta forma que no se podría ver un banco sin un control exacto de sus clientes, una empresa cerrando contratos millonarios de continente a continente en solo unos minutos, ya que es parte fundamental en nuestras vidas tener contacto con las máquinas.

La arquitectura de la informática, es el término general que se aplica a la estructura de un sistema informático o de una parte del mismo. El término se aplica también al diseño del software de sistema, por ejemplo, el sistema operativo, y a la combinación de hardware y software básico que comunica los

aparatos de una red informática. La construcción de ordenadores se refiere a toda una estructura y a los detalles necesarios para que sea funcional, es decir, cubre sistemas informáticos, microprocesadores, circuitos y programas del sistema. Por lo general, el término no suele referirse a los programas de aplicación, como hojas de cálculo o procesadores de textos, que son necesarios para realizar una tarea pero no para que el sistema funcione.

El software, como los lenguajes de programación y los sistemas operativos, hace que los detalles de la arquitectura del hardware resulten invisibles para el usuario. Por ejemplo, diferentes computadoras que empleen el lenguaje de programación C o el sistema operativo UNIX pueden parecer iguales desde el punto de vista del usuario aunque la arquitectura de hardware sea diferente.

Pero no solamente existen las computadoras para grandes trabajos pues los programadores también han creado un software de entretenimiento el cual lo conocemos como videogames (video juegos), existe una gran diversidad de ellos, con el nacimiento de esta diversión se ha cimentado una polémica sobre si son de beneficio o perjudiciales para los usuarios, de un lado hay quien dice que pasar horas enfrente de ellos logra que la persona pierda conciencia de su realidad por la extrema violencia que existe en algunos títulos, por otra parte se ha demostrado que aquellas personas que tienen contacto con este pasatiempo logran crear una habilidad de estar más alertas en el entorno cuando manejan, pues en los juegos se crea una destreza de coordinación que no se desarrollaría con otras labores. De esta manera nos damos cuenta que el avance es necesario para todos los sectores y demandas que seguirán creciendo con los años.

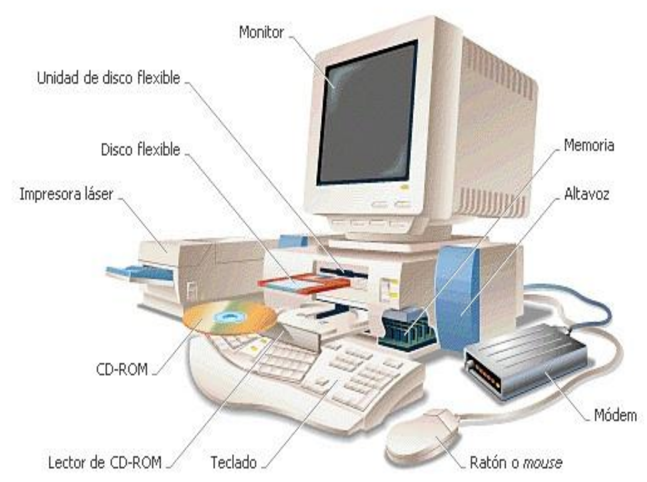
OBJETIVO GENERAL.

Se describirá el desarrollo de la computadora, su múltiple programación así como los diversos lenguajes que controlan el hardware para realizar las tareas indicadas por cada usuario, se ejecutara un breve reconocimiento a las necesidades que han influido en su creación y los beneficios que se han obtenido de los programas informáticos, su aplicación en los ordenadores hasta poder llegar al resultado del diverso software existente, para al final analizar y llevar acabo la elaboración del video juego.

CAPÍTULO I

COMPUTACIÓN EN NUESTROS DIAS.





OBJETIVO.

Se identificarán los tipos de ordenadores, sus características en las cuales se analizarán hardware y software con sus diversas funciones, se conocerán los avances que ha tenido la computación, así como reconocer a uno de los máximos exponentes de software e informática (Microsoft).

Es raro no estar enterado de la computación, pero de su importancia muchas personas no la valoran como debería ser, pues no les importa como funcionan, ni que pasaría si un día no funcionarán las computadoras, o que acontecería si se estancara la tecnología, o que nos a traído de ventajas está Máquina. Todos han podido escuchar de Microsoft, pero de su fundación y de su gran empresario y programador no se escucha mucho, o de el software que proporciona en la actualidad a millones de ordenadores en el mundo, como lo es el caso de Windows, y todas las mejoras recientes de esté sistema operativo.

1.1 ARQUITECTURA DE LA COMPUTACIÓN.

Los computadores constituyen una parte tan importante de la vida moderna que apenas si nos percatamos de su existencia. Pero están en todas partes y parece que sólo nos damos cuenta de su ubicuidad cuando dejan de funcionar. Imaginemos lo que sucedería sin ellos.

1.1.1 LA VIDA SIN LAS COMPUTADORAS.

Usted despierta y advierte que la mañana ya está muy avanzada; el despertador no se ha activado oportunamente; se pregunta si ha dormido de más; tiene un proyecto importante que debe presentar hoy. La carátula de su reloj digital está en blanco. El televisor y la radio no le ayudan, pues no puede hallar una estación en ninguno de ellos. Tampoco puede saber qué hora es por el teléfono, pues éste no funciona.

No se encuentra el diario matutino ante su puerta y tendrá que hacer su propio pronóstico meteorológico mirando por la ventana. No hay música que lo acompañe mientras se viste: su reproductor de discos compactos se rehúsa a atender sus solicitudes. ¿Y qué hay con el desayuno? No puede programar la cafetera automática y el horno de microondas está en huelga.

Así que usted decide salir a desayunar. Su automóvil no enciende; los únicos vehículos que se mueven tienen más de 15 años de antigüedad. Las filas en el metro son interminables. La gente con nerviosismo sobre la avería en el dispositivo de programación del metro, controlado por computador.

Entonces decide entrar a un restaurante de comida rápida y encuentra largas filas de gente esperando mientras los cajeros hacen transacciones a mano. Está hambriento, así que opta por esperar y unirse a la conversación circundante.

Al parecer, todos se interesan en hablar entre sí, dado que no funcionan los instrumentos usuales de comunicación masiva.

Le queda poco dinero en la cartera, así que, después de desayunar, decide hacer una escala en el cajero automático.

¿Para qué?

Vuelve a casa a esperar que llegue el libro que solicitó por correo para entrega en 24 horas. No pasa mucho tiempo antes de darse cuenta de que la espera será larga: los aviones no vuelan por que las instalaciones de control de tráfico aéreo están fuera de servicio. Se dirige a la biblioteca local para ver si el libro se encuentra ahí. Por supuesto, la tarea será ardua porque el catálogo de libros está computarizado.

Mientras camina a casa piensa en las consecuencias de una avería de los computadores a nivel mundial. ¿Cómo se encontrará la gente en los rascacielos de oficinas de alta tecnología, donde los sistemas de computación controlan todo, desde los ascensores hasta el grado de humedad, ¿podrán funcionar las plantas de energía sin el control de computadores? ¿Qué será de los pacientes de los pabellones médicos computarizados? ¿Qué ocurrirá con los satélites que los sistemas de control computarizados mantienen en órbita? ¿Se colapsará la

infraestructura financiera si no hay computadores que procesen y comuniquen las transacciones? ¿Será más seguro el mundo sin las armas controladas por computadoras son inservibles?

Esta historia puede continuar, pero su sentido ya debe de ser obvio. Los computadores son prácticamente ubicuos y nuestra vida es afectada por ellos en muy distintas formas, tanto por su operación como por su falta de funcionamiento. Lo más sorprendente es que los computadores se hayan infiltrado tanto en nuestra vida en un tiempo tan corto.

Las computadoras nos acompañan desde hace apenas medio siglo, pero sus raíces van mucho más allá de la máquina analítica concebida por Babbage. Esta extraordinaria máquina es el producto de siglos de meditación y esfuerzo intelectual.

1.1.2 LA MÁQUINA DE PROCESAMINENTO DE INFORMACIÓN.

Al igual que la máquina analítica, la computadora es quién cambia información de una forma a otra. Todos ellos reciben datos (entrada) y proporcionan información (salida).

Puesto que la información puede presentarse en muchas formas, la computadora es una máquina sumamente versátil, capaz de todo aquello que va desde calcular los impuestos federales sobre la renta hasta guiar los misiles que se compran con esos impuestos. La entrada del ordenador para calcular puede consistir en números de sueldos, otros ingresos, deducciones y tablas fiscales, y la salida puede ser el número que representa los impuestos por pagar. Si la PC se usa para lanzar un misil, las entradas serían señales de radio y radar para localizar el misil y el objetivo; la salida sería una secuencia de señales eléctricas

para controla la trayectoria de vuelo. Lo más sorprendente es que puede usarse la misma computadora para llevar acabo todo esto.

¿Cómo es posible que una máquina sea tan versátil? La flexibilidad del ordenador no está oculta en su hardware (las partes físicas del sistema de cómputo), sino en el software, es decir, en los programas, que son las instrucciones que indican al hardware lo que debe hacer para transformar los datos de entrada en la salida requerida.

Un programa controla todo el proceso, de principio a fin; trátese de un cálculo simple o de la producción de una animación compleja. En efecto, la computadora puede convertirse en una herramienta diferente con sólo cambiar el programa. El típico ordenador moderno es una herramienta de propósito general, ya que puede programarse para realizar distintas tareas.

1.1.3 HARDWARE Y SOFTWARE.

Todos los ordenadores digitales modernos son similares conceptualmente con independencia de su tamaño. Sin embargo, pueden dividirse en varias categorías según su precio y rendimiento: el ordenador o computadora personal es una máquina de costo relativamente bajo y por lo general de tamaño adecuado para un escritorio (algunos de ellos, denominados portátiles, o laptops, son lo bastante pequeños como para caber en un maletín); la estación de trabajo, un microordenador con gráficos mejorados y capacidades de comunicaciones que lo hacen especialmente útil para el trabajo de oficina; el mini ordenador o mini computadora, un ordenador de mayor tamaño que por lo general es demasiado caro para el uso personal y que es apto para compañías, universidades o laboratorios; y el mainframe, una gran máquina de alto precio capaz de servir a las necesidades de grandes empresas, departamentos gubernamentales,

instituciones de investigación científica y similares (las máquinas más grandes y más rápidas dentro de esta categoría se denominan superordenadores).

En realidad, un ordenador digital no es una única máquina, en el sentido en el que la mayoría de la gente considera a los ordenadores. Es un sistema compuesto de cinco elementos diferenciados: una CPU (unidad central de proceso); dispositivos de entrada; dispositivos de almacenamiento de memoria; dispositivos de salida y una red de comunicaciones, denominada bus, que enlaza todos los elementos del sistema y conecta a éste con el mundo exterior.

El CPU puede ser un único chip o una serie de chips que realizan cálculos aritméticos y lógicos y que temporizan y controlan las operaciones de los demás elementos del sistema. Las técnicas de miniaturización y de integración han posibilitado el desarrollo de un chip de CPU denominado microprocesador, que incorpora un sistema de circuitos y memoria adicionales. El resultado son unos ordenadores más pequeños y la reducción del sistema de circuitos de soporte. Los microprocesadores se utilizan en la mayoría de los ordenadores personales de la actualidad.

La mayoría de los chips de CPU y de los microprocesadores están compuestos de cuatro secciones funcionales: una unidad aritmética/lógica; unos registros; una sección de control y un bus interno. La unidad aritmética/lógica proporciona al chip su capacidad de cálculo y permite la realización de operaciones aritméticas y lógicas. Los registros son áreas de almacenamiento temporal que contienen datos, realizan un seguimiento de las instrucciones y conservan la ubicación y los resultados de dichas operaciones. La sección de control tiene tres tareas principales: temporiza y regula las operaciones de la totalidad del sistema informático; su descodificador de instrucciones lee las configuraciones de datos en un registro designado y las convierte en una actividad, como podría ser sumar o comparar, y su unidad interruptora indica en

qué orden utilizará la CPU las operaciones individuales y regula la cantidad de tiempo de CPU que podrá consumir cada operación.

El último segmento de un chip de CPU o microprocesador es su bus interno, una red de líneas de comunicación que conecta los elementos internos del procesador y que también lleva hacia los conectores externos que enlazan al procesador con los demás elementos del sistema informático. Los tres tipos de bus de la CPU son: el bus de control que consiste en una línea que detecta las señales de entrada y de otra línea que genera señales de control desde el interior de la CPU; el bus de dirección, una línea unidireccional que sale desde el procesador y que gestiona la ubicación de los datos en las direcciones de la memoria; y el bus de datos, una línea de transmisión bidireccional que lee los datos de la memoria y escribe nuevos datos en ésta.

Los dispositivos de entrada permiten al usuario del ordenador introducir datos, comandos y programas en la CPU. El dispositivo de entrada más común es un teclado similar al de las máquinas de escribir. La información introducida con el mismo, es transformada por el ordenador en modelos reconocibles. Otros dispositivos de entrada son los lápices ópticos, que transmiten información gráfica desde tabletas electrónicas hasta el ordenador; joysticks y el ratón o mouse, que convierte el movimiento físico en movimiento dentro de una pantalla de ordenador; los scanners luminosos, que leen palabras o símbolos de una página impresa y los traducen a configuraciones electrónicas que el ordenador puede manipular y almacenar; y los módulos de reconocimiento de voz, que convierten la palabra hablada en señales digitales comprensibles para el ordenador. También es posible utilizar los dispositivos de almacenamiento para introducir datos en la unidad de proceso.

Los sistemas informáticos pueden almacenar los datos tanto interna (en la memoria) como externamente (en los dispositivos de almacenamiento). Internamente, las instrucciones o datos pueden almacenarse por un tiempo en los

chips de silicio de la RAM (memoria de acceso aleatorio) montados directamente en la placa de circuitos principal de la computadora, o bien en chips montados en tarjetas periféricas conectadas a la placa de circuitos principal del ordenador. Estos chips de RAM constan de conmutadores sensibles a los cambios de la corriente eléctrica. Los chips de RAM estática conservan sus bits de datos mientras la corriente siga fluyendo a través del circuito, mientras que los chips de RAM dinámica (DRAM, acrónimo de Dynamic Random Access Memory) necesitan la aplicación de tensiones altas o bajas a intervalos regulares aproximadamente cada dos milisegundos para no perder su información.

Otro tipo de memoria interna son los chips de silicio en los que ya están instalados todos los conmutadores. Las configuraciones en este tipo de chips de ROM (memoria de sólo lectura) forman los comandos, los datos o los programas que el ordenador necesita para funcionar correctamente. Los chips de RAM son como pedazos de papel en los que se puede escribir, borrar y volver a utilizar; los chips de ROM son como un libro, con las palabras ya escritas en cada página. Tanto los primeros como los segundos están enlazados a la CPU a través de circuitos.

Los dispositivos de almacenamiento externos, que pueden residir físicamente dentro de la unidad de proceso principal del ordenador, están fuera de la placa de circuitos principal. Estos dispositivos almacenan los datos en forma de cargas sobre un medio magnéticamente sensible, por ejemplo una cinta de sonido o, lo que es más común, sobre un disco revestido de una fina capa de partículas metálicas. Los dispositivos de almacenamiento externo más frecuentes son los disquetes y los discos duros, aunque la mayoría de los grandes sistemas informáticos utiliza bancos de unidades de almacenamiento en cinta magnética. Los discos flexibles pueden contener, según sea el sistema, desde varios centenares de miles de bytes hasta bastante más de un millón de bytes de datos. Los discos duros no pueden extraerse de los receptáculos de la unidad de disco, que contienen los dispositivos electrónicos para leer y escribir datos sobre la

superficie magnética de los discos y pueden almacenar miles de millones de bytes. La tecnología de CD-ROM, que emplea las mismas técnicas láser utilizadas para crear los discos compactos (CD) de audio, permiten capacidades de almacenamiento del orden de varios cientos de megabytes (millones de bytes) de datos.

Los dispositivos de salida permiten al usuario ver los resultados de los cálculos o de las manipulaciones de datos de la computadora. El dispositivo de salida más común es la unidad de visualización (VDU, acrónimo de Video Display Unit), que consiste en un monitor que presenta los caracteres y gráficos en una pantalla similar a la del televisor. Por lo general, las VDU tienen un tubo de rayos catódicos como el de cualquier televisor, aunque los ordenadores pequeños y portátiles utilizan hoy pantallas de cristal líquido (LCD, acrónimo de Liquid Crystal Displays) o electro luminiscentes. Otros dispositivos de salida más comunes son la impresora y el módem. Un módem enlaza dos ordenadores transformando las señales digitales en analógicas para que los datos puedan transmitirse a través de las telecomunicaciones.

Los sistemas operativos internos fueron desarrollados sobre todo para coordinar y trasladar estos flujos de datos que procedían de fuentes distintas, como las unidades de disco o los coprocesadores (chips de procesamiento que ejecutan operaciones simultáneamente con la unidad central, aunque son diferentes). Un sistema operativo es un programa de control principal, almacenado de forma permanente en la memoria, que interpreta los comandos del usuario que solicita diversos tipos de servicios, como visualización, impresión o copia de un archivo de datos; presenta una lista de todos los archivos existentes en un directorio o ejecuta un determinado programa.

Hay un gran abismo entre el ser humano, que tiene una serie de problemas vagos, el frío y rígidamente acotado mundo de la computación.

Aquí es donde interviene el software, que nos permite comunicar a la computadora los problemas y hace posible que estos nos comuniquen las soluciones.

Pero el software moderno de las computadoras no surgió de la nada. Así como el hardware ha evolucionado continuamente para lograr que las PC´s sean más rápidas y potentes, el software también ha mejorado en forma constante para que éstos sean útiles y fáciles de usar.

En cierto sentido, Von Neumann, Eckert y Mauchly establecieron la industria del software, al liberar a los programadores de la tiranía del hardware. En vez de mover conmutadores y conectar cables, los programadores de hoy escriben programas, conjuntos de instrucciones diseñados para resolver problemas, y los alimentan a la memoria de la máquina a través de dispositivos de entrada como el teclado y ratón. Estos programas son el software de la computadora. Como el software se almacena en la memoria, los ordenadores pueden pasar de una tarea a otra y luego regresar a la primera, sin que sea necesario modificar el hardware. Por ejemplo, la computadora sirve como un procesador de textos también puede, si se le instruye debidamente, convertirse en una hoja de cálculo contable, una terminal de telecomunicaciones, un instrumento musical o una máquina de juegos.

¿Qué es el software y cómo puede transformar una masa de circuitos en un camaleón electrónico?

Existen tres categorías principales de software:

- Programas de traducción, con los que los programadores pueden crear otro software.

- Aplicaciones de software, que sirven como herramientas para elevar la productividad de los usuarios de computadores en la resolución de problemas.

- Software de sistema, que coordina las operaciones de hardware y lleva a cabo las tareas ocultas que el usuario de la computadora rara vez observa.

1.1.4 RIESGOS CALCULADOS: MODELADO Y SIMULACIÓN POR COMPUTADOR.

Ya sean parte de una sencilla hoja de cálculo o de un complejo conjunto de ecuaciones, los números muchas veces simbolizan fenómenos del mundo real. El modelado por computadora, es decir, el uso de de ordenadores para crear modelos abstractos de objetos, organismos, organizaciones y procesos, puede efectuarse con hojas de cálculo, aplicaciones matemáticas o lenguajes de programación estándar. La mayoría de las aplicaciones presentadas en este capítulo son ejemplos del modelado por computadora. Un ejecutivo empresarial que crea una hoja de cálculo para proyectar las ganancias y pérdidas trimestrales está intentando modelar el mundo económico que afecta a la compañía. Un ingeniero que usa un procesador matemático para probar la capacidad tensorial de un puente está modelando el puente en forma matemática. Incluso un estadístico que comienza por examinar los datos recabados en el mundo real crea modelos estadísticos para describir los datos.

Los modelos por computador no son siempre algo serio; muchos juegos por computadora son modelos. Los tableros de ajedrez, los juegos de pinball, los campos de batalla, las arenas deportivas, las colonias de hormigas, las ciudades, los calabozos medievales, las culturas interplanetarias y las sociedades mitológicas son modelos en los juegos para PC. En las aulas, los estudiantes usan

modelos de computarizados para viajar por rutas históricas, explorar plantas de energía nuclear, invertir en la bolsa de valores y disecar ranas digitales.

Ya sea que haya sido creado o no para el trabajo, la educación o los juegos, un modelo por computadora es una abstracción: un conjunto de conceptos e ideas diseñados para imitar algún tipo de sistema. Un modelo no es estático; puede ponerse a trabajar en una simulación computarizada para ver cómo opera el modelo en ciertas condiciones. Un modelo bien diseñado debe comportarse como el sistema al cual imita.

Por ejemplo, un ingeniero construye un modelo por computador de un nuevo aeroplano para probar cómo responderá a las órdenes humanas. En una simulación de vuelo corriente, el “piloto” controla el empuje del avión y su ángulo del elevador alimentando datos al modelo del avión. El modelo responde ajustando la velocidad del viento y el ángulo de ascenso y descenso, igual que en un avión real. El piloto responde al nuevo estado de la nave ajustando uno o más controles, a lo cual responde el sistema modificado de nuevo el estado del avión. Este lazo de retroalimentación, en el cual el avión y el piloto reaccionan a los datos del otro, continúa durante toda la simulación.

Un simulador de vuelo puede tener una interfaz gráfica con el usuario, de manera que la pantalla del computador se vea y actúe como el tablero de instrumentos de un avión real y pueda ser manejado interactivamente por pilotos humanos. También podría mostrar simplemente números que representan valores de entrada y salida, y los valores de ingreso pueden ser generados por un piloto simulado, ¡otro modelo por computador! De cualquier manera, suministra gran cantidad de información sobre el comportamiento del avión, siempre y cuando el modelo sea preciso.

Las simulaciones son muy útiles en investigaciones de las ciencias físicas, biológicas y sociales, así como en la ingeniería. Las escuelas, las empresas y las

fuerzas armadas también usan simuladores como forma de instrucción y entrenamiento.

El lado oscuro de la simulación computarizada puede resumirse en seis palabras: ***la simulación no es la realidad.*** El mundo real es un lugar sutil y complejo, y es un reto tremendo capturar incluso una fracción de esa sutileza y complejidad en un simulador por computadora.

1.2 AVANCE CONTINUO.

A partir de 1985, los cambios y avances en la dinámica industria de la computación se vuelven cada vez más veloces y ya no existe mucho sentido clasificar en generaciones como anteriormente se realizaba. Sin embargo, desde esa época se ha hablado mucho de una ***quinta generación*** de computadoras, que se caracteriza por máquinas con inteligencia artificial que entre otras cosas serán capaces de auto mejora, auto programación, auto reparación y recabar experiencias (aprendizaje).

Ya en 1983 se hablaba de esa quinta generación de máquinas, pronosticando su aparición primero para 1990 y después para 1995. Estos ordenadores estarían dotados de lo que se decidió llamar inteligencia artificial. Para llegar a dotar de inteligencia a una computadora, uno de los primeros obstáculos estaba en que el tipo de lenguajes de alto nivel utilizado hasta el momento permitían el procesamiento de datos por medio de algoritmos (es decir, la descripción de los pasos para llegar a un resultado) pero no eran adecuados para procesar conocimiento.

La inteligencia artificial consiste básicamente en hacer que una computadora piense de manera inteligente y al respecto ya se han logrado avances interesantes, aunque la quinta generación todavía está por llegar. La

inteligencia artificial o AI, a partir de sus iniciales en inglés, se basa en estudiar la forma de toma de decisiones y resolver problemas. Estos procesos de pensamiento se descomponen en sus pasos básicos, que después se usan para diseñar programas de computadora que resuelvan esos mismos problemas a través de un proceso similar. Desde luego, esto suena mucho más fácil y sencillo de lo que en realidad es. El asunto de complejidad tal que ha llevado años de trabajo a equipos interdisciplinarios, que incluyen a psicólogos, pedagogos, neurólogos, lingüistas, filósofos y científicos de la computación.

Una de las primeras cosas que descubrieron estos equipos es que sabemos muy poco sobre la verdadera forma en que los seres humanos pensamos, aprendemos y resolvemos problemas, por lo que no es de extrañar que la AI se ocupe, entre otras cosas, de investigar a fondo sobre la representación del conocimiento.

En la actualidad, la inteligencia artificial abarca varios temas, que en una lista breve se podrían anotar como:

- Demostración de teoremas.
- ***Juegos inteligentes.***
- Procesamiento del lenguaje natural.
- Robótica.
- Visión artificial.
- Sistemas expertos.
- Redes neuronales.

Entre estos avances el de la rama de juegos inteligentes ha producido logros tales como la computadora y el programa que vencieron a un campeón mundial de ajedrez, lo cual implica un proceso importante en aspectos de programación, manejo de memoria, aprendizaje y uso de estrategias. La teoría de

juegos, a pesar de su nombre retozón, es un tema importante de las matemáticas aplicadas ya que no es un asunto simple elegir una estrategia ganadora al tomar decisiones en condiciones de incertidumbre.

1.2.1 LAS COMPUTADORAS DENTRO DE DIEZ AÑOS.

Si realizamos una extrapolación a partir de los sistemas actuales, es fácil imaginar una proliferación de visores de alta resolución, desde dispositivos minúsculos de mano o muñeca hasta grandes pantallas empotradas en escritorios, paredes y pisos. Estos visores, sin duda, llegarán a ser algo común. Pero muchos científicos de la computación creen que una clase fundamentalmente diferente de interfase de usuario, conocida como realidad aumentada (RA), tendrá un efecto más profundo en la forma como desarrollemos e interactuemos con las futuras computadoras.

La RA se refiere a los visores de cómputo que añaden información virtual a las percepciones sensoriales del usuario. La mayoría de la investigación en este campo se enfoca a dispositivos “transparentes”, utilizados usualmente en la cabeza, que sobreponen gráficos y textos sobre la visión del entorno del usuario. (La información virtual también puede estar en otras formas sensoriales, como sonido o tacto, pero en este artículo nos concentramos en el incremento visual.) Los sistemas de RA rastrean la posición y orientación de la cabeza del usuario, de modo que al material sobrepuesto se pueda alinear con la visión global del usuario. Mediante este proceso conocido como registro, el software gráfico puede colocar una imagen tridimensional de una taza de té, por ejemplo, sobre un platillo real y mantener la taza virtual fija en esa posición, conforme el usuario se mueve en la habitación. Los sistemas de RA utilizan parcialmente las mismas tecnologías de hardware en la investigación de la realidad virtual, pero con una diferencia crucial: mientras la realidad virtual aspira descaradamente a reemplazar el mundo real, la realidad aumentada respetuosamente lo suplementa.

Un jugador de computadora podría combatir con extraterrestres de tres metros de altura mientras camina hacia el trabajo.

1.2.2 AVANCES INFORMÁTICOS.

Con el desarrollo y evolución de las tecnologías se ven incrementadas las potencialidades educativas. El rápido avance tecnológico de soportes informáticos, como los ordenadores (computadoras), los discos de vídeo digital y los discos compactos, permite el uso de mejores herramientas para profesores y alumnos en el ámbito de la educación. Los discos compactos (el CD-ROM y el CD-I) se utilizan para almacenar grandes cantidades de datos, como enciclopedias universales y especializadas o películas sobre cualquier tema de interés. Con estos nuevos equipos informáticos interactivos, un estudiante interesado en cualquier materia podrá consultar el texto en una enciclopedia electrónica, ver además fotografías o una película sobre el tema, o buscar asuntos relacionados con sólo presionar un botón. Estos soportes tienen la ventaja de que ofrecen la posibilidad de combinar textos con fotografías, ilustraciones, vídeos y audio para ofrecer una visión más completa, además de que presentan una gran calidad. Con los últimos avances tecnológicos, aún en desarrollo, la enseñanza y el aprendizaje comienzan a ser tareas gratas e, incluso, divertidas.

Si bien es cierto que trasladar a los jóvenes desde su papel pasivo de receptores, al activo de intérpretes y creadores no es fácil, también es cierto que gran parte de la tecnología actual, como los materiales multimedia, los vídeos y juegos electrónicos didácticos, puede convertirse en un verdadero aliado de la enseñanza.

Otro de los logros ha sido los métodos empleados en la previsión del tiempo, han experimentado una serie de cambios rápidos desde la II Guerra Mundial en respuesta a los avances en la tecnología de los equipos informáticos,

los satélites y las comunicaciones. Las investigaciones prosiguen con el mismo ímpetu, por lo que cabe esperar que se produzcan muchos más cambios en la próxima década.

Un gran progreso es la micro miniaturización este circuito integrado, un microprocesador F-100, tiene sólo 0,6 cm2, y es lo bastante pequeño para pasar por el ojo de una aguja.A. Sternberg/Ferranti lgor. /Science Source/Photo Researchers, Inc.

Una tendencia constante en el desarrollo de los ordenadores es la micro miniaturización, iniciativa que tiende a comprimir más elementos de circuitos en un espacio de chip cada vez más pequeño. Además, los investigadores intentan agilizar el funcionamiento de los circuitos mediante el uso de la superconductividad, un fenómeno de disminución de la resistencia eléctrica que se observa cuando se enfrían los objetos a temperaturas muy bajas.

Las redes informáticas se han vuelto cada vez más importantes en el desarrollo de la tecnología de computadoras. Las redes son grupos de computadoras interconectados mediante sistemas de comunicación. La red pública Internet es un ejemplo de red informática planetaria. Las redes permiten que las computadoras conectadas intercambien rápidamente información y, en algunos casos, compartan una carga de trabajo, con lo que muchas computadoras pueden cooperar en la realización de una tarea. Se están desarrollando nuevas tecnologías de equipo físico y soporte lógico que acelerarán los dos procesos mencionados.

Otra tendencia en el desarrollo de computadoras es el esfuerzo para crear computadoras de quinta generación, capaces de resolver problemas complejos en formas que pudieran llegar a considerarse creativas. Una vía que se está explorando activamente es el ordenador de proceso paralelo, que emplea muchos chips para realizar varias tareas diferentes al mismo tiempo. El proceso paralelo

podría llegar a reproducir hasta cierto punto las complejas funciones de realimentación, aproximación y evaluación que caracterizan al pensamiento humano. Otra forma de proceso paralelo que se está investigando es el uso de computadoras moleculares. En estas computadoras, los símbolos lógicos se expresan por unidades químicas de ADN en vez de por el flujo de electrones habitual en las computadoras corrientes. Las computadoras moleculares podrían llegar a resolver problemas complicados mucho más rápidamente que las actuales supercomputadoras y consumir mucha menos energía.

1.3 MICROSOFT.

Microsoft Corporation, compañía estadounidense líder en el mercado de software para ordenadores o computadoras. Desarrolla y vende una amplia gama de productos de software tanto a organizaciones como a particulares de más de 50 países. Los sistemas operativos Windows de Microsoft son los que más se utilizan en todo el mundo. La sede de la compañía se encuentra en Redmond, Washington, Estados Unidos.

Entre otros productos de Microsoft mundialmente conocidos se encuentran el procesador de textos Word; la hoja de cálculo Excel; Access, un programa de base de datos, y PowerPoint, un programa para hacer presentaciones corporativas. Estos programas se pueden adquirir por separado o como parte de Office, un paquete integrado de programas informáticos. La compañía también desarrolla BackOffice, un paquete de soluciones de servidores para empresas. Internet Explorer permite al usuario navegar por World Wide Web. Entre la amplia gama de productos también se incluyen los de consulta, como la Enciclopedia Encarta; los juegos; programas contables y financieros; lenguajes de programación, y dispositivos de entrada de datos, como los teclados y los dispositivos señaladores, así como libros sobre informática.

Microsoft gestiona The Microsoft Network (MSN), una colección de sitios Web con información de actualidad, financiera y de ocio. Microsoft y la National Broadcasting Company (NBC) operan conjuntamente MSNBC, una cadena de televisión por cable que emite noticias y coloquios las 24 horas del día y que además tiene página en Internet.

1.3.1 FUNDACIÓN.

La compañía fue fundada en 1975 por William H. Gates III y Paúl Allen. Ambos se habían conocido durante su época de estudiantes por su afición común a programar con la computadora PDP-10 de Digital Equipment Corporation. En ese año, la revista Popular Electronics dedicó su portada y un artículo al Altair 8800, la primera computadora personal. El artículo animó a Gates y Allen a desarrollar la primera versión del lenguaje de programación BASIC para el equipo Altair. Le compraron la licencia de este software a Micro Instrumentation and Telemetry Systems (MITS), la empresa fabricante del Altair, y fundaron Microsoft (originalmente Micro-soft) en Albuquerque, Nuevo México, con el objeto de desarrollar versiones de BASIC para otras compañías del sector. Apple Computer, fabricante del equipo Apple II, Commodore, fabricante del PET, y Tandy Corporation, fabricante del equipo Radio Shack TRS-80, todas ellas compañías creadas por aquel entonces, fueron los primeros clientes de Microsoft. En 1977 Microsoft lanzó al mercado su segundo producto, Microsoft FORTRAN, otro lenguaje de programación, y pronto sacó versiones del lenguaje BASIC para los microprocesadores 8080 y 8086.

1.3.2 MS-DOS.

En 1979 Gates y Allen trasladaron la compañía a Bellevue, Washington, muy cerca de Seattle, la ciudad donde ambos nacieron. El traslado a Redmond, muy próximo a Bellevue, se produjo en 1986. En 1980 IBM contrató a Microsoft

para escribir el sistema operativo del IBM PC, que saldría al mercado al año siguiente. Presionada por el poco tiempo disponible, Microsoft compró QDOS (Quick and Dirty Operating System) a Tim Paterson, un programador de Seattle, por 50.000 dólares y le cambió el nombre a MS-DOS. El contrato firmado con IBM permitía a Microsoft vender este sistema operativo a otras compañías. En 1984 Microsoft había otorgado licencias de MS-DOS a 200 fabricantes de equipos informáticos y, así, este sistema operativo se convirtió en el más utilizado para PC, lo que permitió a Microsoft crecer vertiginosamente en la década de 1980.

El comando indicador (C:\>) señala que está listo para aceptar cualquier comando. Se digita un comando para ejecutar una tarea o para iniciar un programa. Un comando simple puede identificarle a su computadora lo que desea lograr. Por ejemplo: el comando date le dice a la computadora que muestre la fecha actual. El cursor es la línea intermitente en la pantalla. Este le indica dónde aparecerá el texto que digita.

Como los archivos en un archivador, MS-DOS utiliza directorios para organizar datos almacenados en una computadora. El directorio madre es C: todos los demás se encuentran dentro de este.

Cuando almacena un archivo en la computadora, debe de asignarle un nombre. Este nombre no debe incluir espacios y consiste en una palabra con su extensión, separado por un punto. El nombre describe el contenido del archivo y puede tener hasta ocho caracteres. La extensión identifica el tipo de archivo y consiste en tres caracteres.

Nota: *MS-DOS significa Sistema Operativo en Disco de Microsoft.*

1.3.3 SOFTWARE PARA APLICACIONES.

A medida que las ventas de MS-DOS se disparaban, Microsoft empezó a desarrollar una serie de aplicaciones para PC con fines comerciales. En 1982 salió al mercado Multiplan, un programa de hoja de cálculo, y el año siguiente se puso a la venta el procesador de textos denominado Microsoft Word. En 1984 Microsoft fue una de las primeras compañías del sector que se dedicó a desarrollar aplicaciones para Macintosh, una computadora personal creada por la compañía Apple Computer. En un principio, Microsoft obtuvo grandes éxitos de venta de programas para Macintosh como Word, Excel y Works (un grupo de aplicaciones integradas en un paquete). No obstante, Multiplan para MS-DOS fue casi totalmente sustituido por la famosa hoja de cálculo de Lotus Development Corporation, Lotus 1-2-3.

1.3.4 WINDOWS.

En 1985 Microsoft lanzó Windows, un sistema operativo que ampliaba las prestaciones de MS-DOS e incorporaba por primera vez una interfaz gráfica de usuario. Windows 2.0, que salió a la venta en 1987, mejoraba el rendimiento y ofrecía un nuevo aspecto visual. Tres años más tarde apareció una nueva versión, Windows 3.0, a la que siguieron Windows 3.1 y 3.11. Estas versiones, que ya venían preinstaladas en la mayoría de los equipos, se convirtieron rápidamente en los sistemas operativos más utilizados de todo el mundo. En 1990 Microsoft pasó a ser la empresa líder de programas informáticos y alcanzó unas ventas anuales de más de mil millones de dólares.

Cuando Microsoft se encontraba en la cima del mercado de los programas para PC, la compañía fue acusada de ejercer prácticas empresariales monopolísticas. En 1990, la Comisión Federal de Comercio estadounidense (FTC, siglas en inglés) comenzó a investigar a Microsoft por supuestas prácticas

contrarias a la libre competencia, pero fue incapaz de dictar sentencia y cerró el caso. El Departamento de Justicia estadounidense continuó la investigación.

En 1991 Microsoft e IBM finalizaron una década de colaboración cuando decidieron seguir caminos diferentes en la siguiente generación de sistemas operativos para ordenadores personales. IBM continuó con un antiguo proyecto en común con Microsoft, un sistema operativo denominado OS/2 (que salió al mercado en 1987), mientras Microsoft decidió desarrollar su sistema operativo gráfico Windows. En 1993 Apple perdió un juicio contra Microsoft al que acusaba de violación de las leyes de derechos de autor por haber copiado ilegalmente el diseño de la interfaz gráfica de Macintosh. El fallo fue más adelante confirmado por un tribunal de apelación.

Windows NT, un sistema operativo diseñado para entornos empresariales, fue lanzado en 1993. Al año siguiente, la compañía y el Departamento de Justicia firmaron un acuerdo en el que se pedía a Microsoft que modificase la forma de vender y conceder licencias para sus sistemas operativos a los fabricantes de computadoras. En 1995 la compañía lanzó Windows 95, un entorno multitarea con interfaz simplificada y con otras funciones mejoradas. A las siete semanas de su lanzamiento se habían vendido siete millones de copias.

1.3.4.1 WINDOWS 3.1

Windows 3.1 despliega imágenes en la pantalla para ayudarle a ejecutar tareas. El administrador de programas es el centro de control donde puede abrir programas.

El icono le permite iniciar un programa, tal como un procesador de palabras. Un icono es una imagen pequeña que representa a un programa. Una ventana posee una barra de título que despliega el nombre de la aplicación.

El administrador de archivos le permite ver y organizar todos los archivos almacenados en su computadora. Windows 3.1 usa directorios para la organizar información, del mismo modo que usarían carpetas para organizar los papeles en un archivero. El escritorio es el área del fondo de la pantalla.

1.3.4.2 WINDOWS 95.

Windows 95 es un verdadero sistema operativo ya que no requiere de MS-DOS para funcionar. Mi PC le permite navegar a través de todas las carpetas y documentos almacenados en su computadora. La papelera de reciclaje almacena documentos que ha eliminado y le permite recuperarlos más tarde. Entorno de red permite ver carpetas y archivos disponibles en su red. Con el botón inicio le permite acceder rápidamente a programas y documentos. La barra de tareas contiene el botón inicio y muestra el nombre de cada ventana abierta. Windows Explorer le muestra el sitio de cada carpeta y documento de su ordenador, puede usarlo para mover, abrir, imprimir o eliminar documentos. Existen accesos directos que brinda una forma más rápida de abrir un documento regularmente.

1.3.4.3 WINDOWS 98.

Actualmente, hay dos versiones de Windows 98. Windows 98 segunda Edición incluye muchas mejoras y actualizaciones para la versión original de Windows 98. Puede revisar su disco CD-ROM de Windows 98 segunda edición instalado en su computadora.

Mis documentos brinda un lugar conveniente para almacenar archivos personales; el Explorador de Internet le permite navegar a través de la información en la World Wide Web; la barra de herramientas Quick Launch le permite acceder a las características más usadas, incluyendo Internet Explorer y Outlook Express.

Windows 98 es más confiable que Windows 95 e incluye muchas herramientas que puede utilizar para encontrar y solucionar problemas con su computadora, por ejemplo, puede buscar errores en su disco duro, eliminar archivos innecesarios y desfragmentar el disco duro para mejorar su desempeño.

1.3.4.4 WINDOWS ME.

Windows ME significa Millennium Edition, es el sucesor de Windows 98, ya que esta familiarizado con la forma se ve y funciona; Windows ME le permite grabar, editar y guardar videos en su computadora utilizando Windows Movie Marker. Una vez que haya terminado de trabajar con el video, puede enviarlo por correo electrónico a sus amigos y/o colocarlo en una página Web; esta versión posee una mejora de Windows Media Player que le ayudará a administrar sus archivos multimedia y escuchar estaciones de radio por Internet.

Si experimenta problemas con su computadora, puede utilizar System Restore para devolver su computadora a un momento anterior a que ocurriera los problemas; por ejemplo: si borramos un programa accidentalmente, puede restaurar la computadora antes de que eliminara los archivos.

1.3.4.5 WINDOWS NT.

Windows NT Workstation es una versión que usa en redes clientes/servidor y en algunas redes de persona a persona. Muchas aplicaciones poderosas son diseñadas específicamente para operar en este sistema operativo, muchos programas diseñados para versiones anteriores funcionarán, y mejor, en Windows NT Workstation.

Windows NT Server es una versión del sistema operativo que usa redes de cliente/servidor; esta diseñado para dar soporte a las demandas pesadas de

procesamiento de un servidor de la red. Las computadoras cliente en una red operando con Windows NT Server pueden usar una variedad de sistemas operativos, como Linux, Windows 98 y Macintosh Os9.

1.3.4.6 WINDOWS 2000.

Windows 2000 profesional se usa comúnmente en redes de cliente/servidor pero también se puede usar en redes de persona a persona. Este sistema operativo ofrece una estabilidad incrementada y proporciona las herramientas que le ayudan a darle mantenimiento a su computadora, esta versión está diseñada para uso empresarial. Windows 2000 Server y Windows 2000 Advanced Server se encuentran en grandes redes de cliente/servidor. Estos dos sistemas operativos están diseñados para dar soporte a grandes demandas de procesos de redes. Windows Advanced Server puede dar soporte a una red incluso más grande que Windows 2000 Server, lo cual hace a la versión Advanced Server la más conveniente para las grandes empresas y proveedores de servicio de Internet.

1.3.4.7 WINDOWS XP.

Se observará algunos cambios tan pronto como inicie Microsoft Windows XP. El fondo del escritorio, el menú Inicio y la barra de herramientas tienen un aspecto diferente. Pero las características nuevas o notablemente a la presentación. No admitirá algunos de los cambios hasta que comience a utilizar el programa.

Es el sucesor de Windows 2000 y Windows Millenium y se trata del primer sistema operativo diseñado expresamente por Microsoft para responder a las necesidades tanto de usuarios individuales como de compañías.

El sistema operativo funciona como el cerebro y el sistema nervioso de su equipo; es decir, índica al equipo que debe de hacer y cómo hacerlo. El sistema interactúa con software instalados en el equipo y permite comunicar al ordenador con otros equipos y componentes de hardware.

1.3.5 MEJORAS RECIENTES.

Microsoft empezó a operar en el campo de los medios de comunicación y creó The Microsoft Network en 1995 y MSNBC un año después. Además, en 1996 Microsoft presentó Windows CE, un sistema operativo para computadoras de bolsillo. En 1997 Microsoft pagó 425 millones de dólares por la adquisición de WebTV Networks, un fabricante de dispositivos de bajo costo para conectar televisiones a Internet. Ese mismo año Microsoft invirtió mil millones de dólares en Comcast Corporation, un operador estadounidense de televisión por cable, como parte de su política de extender la disponibilidad de conexiones de alta velocidad a Internet.

A finales de 1997 el Departamento de Justicia acusó a Microsoft de violar el acuerdo de 1994 al obligar a los fabricantes de computadoras que instalaban Windows 95 a la inclusión de Internet Explorer, un programa para navegar por Internet. El gobierno sostuvo que la compañía se estaba aprovechando de su posición en el mercado de los sistemas operativos para obtener el control de los exploradores de Internet. Microsoft afirmó que debía disponer del derecho de mejorar la funcionalidad de Windows integrando en el sistema operativo funciones relacionadas con Internet. Por otra parte, a finales de 1997, la compañía Sun Microsystems demandó a Microsoft, alegando que había incumplido el contrato por el que se le permitía utilizar el lenguaje de programación universal JAVA, de Sun, al introducir mejoras sólo para Windows. En noviembre de 1998 un tribunal dictó sentencia en contra de Microsoft por un mandamiento presentado por Sun ese mismo año. Dicho mandamiento obligaba a Microsoft a revisar su software

para cumplir los estándares de compatibilidad con JAVA. Microsoft apeló la sentencia.

A principios de 1998 Microsoft llegó a un acuerdo temporal con el Departamento de Justicia que permitía a los fabricantes de PC ofrecer una versión de Windows 95 sin acceso a Internet Explorer. No obstante, en mayo de1998 el Departamento de Justicia y veinte estados de Estados Unidos presentaron demandas contra Microsoft por supuestas prácticas monopolísticas y por abusar de posición dominante en el mercado para hundir a la competencia. Estas demandas obligaron a Microsoft a vender Windows sin Internet Explorer o a incluir Navigator, el explorador de Web de Netscape Communications Corporation, la competencia. Las demandas también obligaron a modificar algunos contratos y la política de precios.

En junio de 1998 Microsoft lanzó Windows 98, que incluye funciones integradas para acceso a Internet. Al mes siguiente Bill Gates nombró presidente de la compañía a Steve Ballmer, hasta entonces vicepresidente ejecutivo, y éste se hizo cargo de la supervisión de las actividades empresariales cotidianas de Microsoft.

El juicio contra Microsoft por haber violado las leyes antimonopolio comenzó en octubre de 1998. Ejecutivos de Netscape, Sun y otras muchas compañías de software y hardware testificaron acerca de sus contratos empresariales con Microsoft. En noviembre de 1999 el juez del tribunal federal, tras escuchar a los testigos convocados, expuso sus conclusiones y declaró que Microsoft poseía un monopolio en el mercado de sistemas operativos. En abril de 2000, el juez hizo pública la sentencia contra la compañía por haber violado las leyes antimonopolio al emplear tácticas que entorpecían la competitividad. Microsoft recurrió el fallo.

En 1999 Microsoft pagó 5.000 millones de dólares a la compañía de telecomunicaciones AT&T Corp. Para utilizar su sistema operativo Windows CE en

dispositivos diseñados para ofrecer a los consumidores servicios integrados de televisión por cable, teléfono y acceso rápido a Internet. Además, ese mismo año, la compañía lanzó Windows 2000, la versión más actualizada del sistema operativo Windows NT. En enero de 2000 Bill Gates traspasó su cargo de presidente ejecutivo (CEO) a Ballmer, un cambio que le ha permitido a aquél centrarse en el desarrollo de nuevos productos y tecnologías.

1.3.6 WILLIAM HENRY GATES III.

William Henry Gates III (1955-), empresario estadounidense, creador de Microsoft Corporation, compañía líder en el mercado de software para ordenadores o computadoras en Estados Unidos, que fundó en 1975 con su compañero de estudios Paúl Allen. El éxito de dicha compañía convirtió a Gates en una de las figuras más influyentes de la industria informática y, con el tiempo, en el hombre más rico del mundo.

Nació en Seattle (Washington) y asistió a la escuela pública hasta el sexto grado. Cursó el séptimo en el Lakeside School de su ciudad natal, centro en el que conoció a Allen. Gates empezó a trabajar con computadoras y lenguajes de programación en 1968, cuando cursaba octavo grado. Ese mismo año, el Lakeside School adquirió un teletipo que se conectaba a un ordenador central a través de las líneas telefónicas. En aquel momento, Lakeside School era uno de los pocos colegios que disponía de computadoras para uso de los alumnos.

Poco tiempo después, Gates, Allen y otros alumnos convencieron a una compañía informática local para que les proporcionara libre acceso al PDP-10, una nueva mini computadora desarrollada por Digital Equipment Corporation. A cambio de poder utilizarla, intentarían encontrar errores en el sistema. Gates pasó gran parte de su tiempo libre estudiando el PDP-10 y aprendiendo lenguajes de programación tales como BASIC, FORTRAN y LISP. En 1972 Gates y Allen

fundaron Traf-O-Data, una compañía dedicada al diseño y fabricación de dispositivos computerizados de recuento de automóviles para analizar el tráfico. Cuando trabajaban en este proyecto conocieron el microprocesador 8008 programable de Intel Corporation.

En 1975, mientras estudiaba en la Universidad de Harvard, Gates se asoció con Allen para desarrollar una versión del lenguaje de programación BASIC para el Altair 8800, la primera computadora personal. Licenciaron este software a Micro Instrumentation and Telemetry Systems (MITS, la empresa fabricante del Altair 8800) y fundaron Microsoft (originalmente Micro-soft) con el objetivo de desarrollar versiones de BASIC para otras compañías del sector. Gates decidió abandonar Harvard en el tercer año de sus estudios universitarios para dedicarse por completo a Microsoft. En 1999 su fortuna personal sobrepasaba los 80.000 millones de dólares, lo que le convertía en el hombre más rico del mundo. Ese mismo año donó más de 2.400 millones de dólares a obras benéficas.

Gates también ha realizado inversiones personales en otras compañías dedicadas a la alta tecnología. En 1989 fundó Corbis Corporation, propietaria en la actualidad de la mayor colección de imágenes digitales del mundo. En 1995 Corbis adquirió el Bettmann Archive, integrado por más de 16 millones de imágenes fotográficas y anunció su intención de digitalizar parte de la colección. En 1994, Gates y Craig McCaw, pionero en la industria de las telecomunicaciones celulares, pasaron a ser los principales inversores de Teledesic Corporation. Teledesic planeaba lanzar varios centenares de satélites artificiales de órbita baja con el objeto de crear una red de telecomunicaciones global y de alta velocidad. Gates expuso su visión del papel que juega la tecnología en la sociedad en su libro Camino al futuro (1995). En 1999 publicó Los negocios en la era digital, en cuyas páginas incidió en la importancia de la tecnología en los negocios.

En 1998 nombró presidente de la compañía a Steve Ballmer, hasta entonces vicepresidente ejecutivo, aunque él siguió ocupando el cargo de

presidente ejecutivo (CEO). En 2000 transfirió también este último a Ballmer, lo que le permitió centrarse en el desarrollo de nuevas tecnologías y productos.

Como cierre del capítulo se ha podido explicar brevemente como funciona una computadora, de la gran importancia a nivel mundial para múltiples labores realizadas por estás máquinas hoy en día. Se hablo de Microsoft de tal forma que se reviso su fundación, a su dueño, así como los diferentes sistemas operativos que han salido al mercado a partir de MS-DOS sistema de comandos hasta llegar a Windows Xp. Además se tomo el tema de que los avances son y deberán seguir siendo continuos, de tal forma que cada día se está pensando y creando nuevas ideas para mejorar las computadoras y sus sistemas.

CAPÍTULO II

PROGRAMACIÓN Y SUS APLICACIONES ACTUALES

```
Símbolo del sistema - TC
     File      Edit      Run      Compile      Project      Options      Debug
                                   Edit
      Line 11     Col 10   Insert Indent Tab A:NONAME.C
#INCLUDE"STDIO.H"
#INCLUDE"CONIO.H"
#INCLUDE"GRAPHICS.H"

VOID MAIN()
INT A,B,C,M,S;
{
GOTOXY(10,15);
PRINTF("SELECCIONA EL LA OPCIóN DE JUEGO");
SCANF(%D,&A¿?);
GETCHE();
[
                                  Message

F1-Help  F5-Zoom  F6-Message  F9-Make  F10-Main menu              CAPS  NUM
```

OBJETIVO.

Se determinará la evolución de los lenguajes de programación, con la intención de identificar las diferencias entre ellos, su nivel de lenguaje, además se considerara su aplicación en la actualidad.

La estructura de una computadora es un tanto compleja, ya que para que funcione está máquina no es lo más importante tener el mejor hardware en el computador, pues si no existe un software que logre controlar al ordenador en el que se trabaje, el aparato es totalmente inútil, por lo cual es importante saber que tipo de códigos y lenguajes existen, los tipos de programación y su uso en la actualidad mundial como es el caso de la automatización.

2.1 CODIGOS Y LENGUAJES.

Por programar se entiende un proceso mental complejo, dividido en varias etapas. La finalidad de la programación, así entendida, es comprender con claridad la situación o el programa que se va a simular o resolver por medio de la computadora, y encender también con detalle cuál será el procedimiento mediante el cual la máquina llegará a la solución deseada.

La codificación constituye una etapa necesariamente posterior a la sistematización, y consiste en describir, en un lenguaje de programación adecuado, la solución ya encontrada o sugerida, por medio de la programación. Es decir, primero se programa la solución de un problema y después se traduce para la computadora.

La actividad de programar es más que nada conceptual, y su finalidad es intentar definir, cada vez con mayor precisión, acercamientos para resolver el problema de manera virtual, efectuando una especie de “experimentos mentales” sobre la situación a simular o el problema por resolver. El resultado de tales experimentos constituirá una descripción de lo que se requiere para resolver la solución.

Sin embargo, en situaciones o problemas complejos, un solo programa no es suficiente para modelar adecuadamente la realidad, por lo que previo a la

programación (y mucho antes de la codificación) es necesario realizar un diseño completo, que servirá de guía para la tarea.

Existen varios métodos para obtener esto: tan sólo uno de ellos, conocido como diseño estructurado o descendente, y la correspondiente programación estructurada y modular. Una versión "más moderna" y amplia es el llamado diseño orientado a objetos, y es correspondiente programación orientada a objetos (POO).

Lenguajes de programación

LENGUAJE	ORIGEN DEL NOMBRE	AÑO	USOS/COMENTARIOS
ADA	Augusta ADA Byron (Lady Lovelace)	1979	Derivado de Pascal, utilizado principalmente por los militares.
ALGOL	lgorithmic Language (Lenguaje ALGOL algorítmico)	1960	Primer lenguaje de programación procedural estructurado, utilizado sobre todo para resolver problemas matemáticos.
APL	A Programming Language (Un lenguaje de programación)	1961	Lenguaje interpretado que utiliza un amplio conjunto de símbolos especiales y que se caracteriza por su brevedad. Utilizado fundamentalmente por los matemáticos.
BASIC	Beginners All-Purpose Symbolic Instruction Code(Código de instrucciones simbólicas multipropósito para principiantes)	1965	Lenguaje de programación de alto nivel, utilizado con frecuencia por programadores principiantes.
C	Predecesor del lenguaje de programación B, fue desarrollado en Bell Laboratory, en 1972	1972	Lenguaje de programación compilado y estructurado, que suele utilizarse en numerosos lugares de trabajo porque sus programas pueden transferirse fácilmente entre distintos tipos de computadoras.
COBOL	Common Business-Oriented Language (Lenguaje simbólico de programación orientado a aplicaciones comerciales)	1959	Lenguaje de programación semejante al idioma inglés, que hace hincapié en las estructuras de datos. De amplia utilización, principalmente en empresas.

FORTH	Lenguaje de cuarta (FOuRTH) generación	1970	Lenguaje estructurado e interpretado de fácil ampliación. Ofrece una alta funcionalidad en un espacio reducido.
FORTRAN	FORmula TRANslation (Traducción de fórmulas)	1954	Diseñado en un principio para usos científicos y de ingeniería, se trata de un lenguaje compilado de alto nivel que hoy se utiliza en numerosos campos. Precursor de diversos conceptos, como las variables, las instrucciones condicionales y las subrutinas compiladas por separado.
LISP	LISt Processing (Procesamiento de listas)	1960	Lenguaje de programación orientado a la generación de listas, utilizado principalmente para manipular listas de datos. Lenguaje interpretado que suele utilizarse en las investigaciones y está considerado como el lenguaje estándar en proyectos de inteligencia artificial.
LOGO	Derivado del griego *logos*, 'palabra'.	1968	Lenguaje de programación que suele utilizarse con niños. Presenta un sencillo entorno de dibujo y varias prestaciones de mayor nivel del lenguaje LISP. Fundamentalmente educativo.
Modula-2	MODUlar Language-2, diseñado como fase secundaria de Pascal (diseñados ambos por Niklaus Wirth)	1980	Lenguaje que hace hincapié en la programación modular. Es un lenguaje de alto nivel basado en Pascal, que se caracteriza por la ausencia de funciones y procedimientos estandarizados.
Pascal	Blaise PASCAL, matemático e inventor del primer dispositivo de computación.	1971	Lenguaje compilado y estructurado basado en ALGOL. Agrega tipos y estructuras de datos simplificando la sintaxis. Al igual que el C, se trata de un lenguaje de programación estándar para microcomputadoras.
PILOT	Programmed Inquiry, Language Or Teaching (Consulta, lenguaje o aprendizaje de investigación programada)	1969	Lenguaje de programación utilizado fundamentalmente para crear aplicaciones destinadas a instrucciones asistidas por computadora. Se caracteriza por utilizar un mínimo de sintaxis.
PL/1	Programming Language 1 (Lenguaje de	1964	Diseñado para combinar las principales virtudes del FORTRAN, COBOL y ALGOL, se trata de un lenguaje de

	programación uno)	programación complejo. Compilado y estructurado, es capaz de gestionar errores y de procesar multitareas, y se emplea en entornos académicos y de investigación.

2.1.1 ¿QUÉ ES UN PROGRAMA?

Un programa es un conjunto de instrucciones que controlan (dirigen) a una computadora; más fundamentalmente, un programa de computador es un conjunto de instrucciones internas utilizadas para ejecutarse en el ordenador y produzca un resultado concreto. El paso de escribir un programa o software, se denomina programación y el conjunto de instrucciones que se utiliza para su construcción se les llama lenguajes de programación.

Por ejemplo, un procesador de textos es un programa ejecutable (.exe) que se encarga de recibir el texto como entrada de información, procesándolo de acuerdo con los requerimientos del usuario y generar una salida que pueda ser impresa o vista en pantalla.

En comienzo, la programación era rudimentaria, se basaba en tarjetas perforadas con códigos que activaban interruptores y circuitos. Ahora, la programación ha evolucionado de tal manera que el software para programar es capaz de generar los códigos sin que el programador tenga que perder tiempo en escribirlos.

Una de las características de un programa ejecutable es que está diseñado para que las computadoras lo descifren y no el usuario final, el cual sólo debe preocuparse por saber utilizar la interfaz del programa.

Un programador es la persona que se encarga de crear el código fuente mediante un procesador de texto o una ventana de edición, generalmente incluida

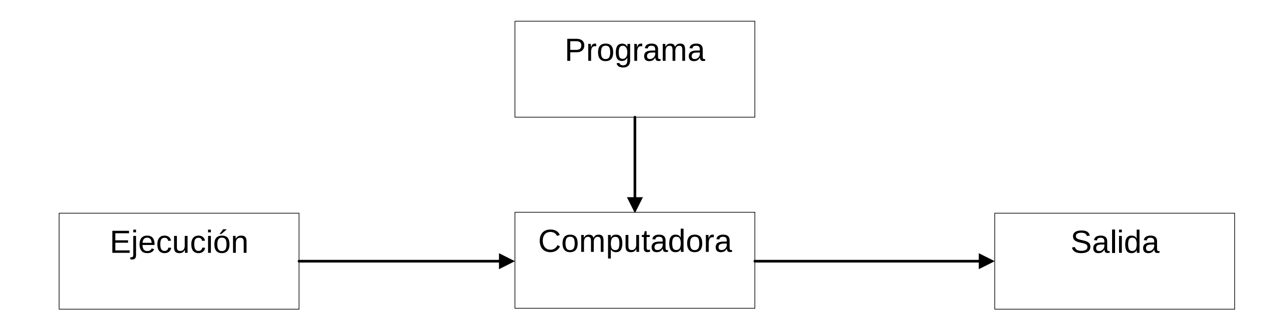
en los lenguajes de programación. El programa está constituido por una serie de líneas de en la que aparecen comandos (palabras reservadas del lenguaje de programación), operadores, instrucciones, variables y constantes. La función del programador es utilizar todas estas herramientas, enlazarlas y crear un programa ejecutable con un propósito específico.

Cuando un programa se ejecuta (run) normalmente hay dos tipos de entrada a una computadora.

El software es un tipo de entrada (contiene instrucciones para el ordenador las realice).

El otro tipo de entrada son los datos del programa de computador.

Es la información que procesará la máquina. Por ejemplo, si es un software de corrección de textos, los datos como el propio programa son entradas, la salida es el resultado producido por el ordenador.



Los lenguajes de programación permiten la comunicación usuario/máquina. Unos programas traductores llamados traductores (compiladores o intérpretes) convierten las el lenguaje de programación a instrucciones escritas en lenguaje máquina (0 y 1, bits) que esta pueda entender.

Lenguaje de programación, en informática, cualquier lenguaje artificial que puede utilizarse para definir una secuencia de instrucciones para su procesamiento por un ordenador o computadora. Es complicado definir qué es y qué no es un lenguaje de programación. Se asume generalmente que la traducción de las instrucciones a un código que comprende la computadora debe ser completamente sistemática. Normalmente es la computadora la que realiza la traducción.

2.1.2 LENGUAJE MÁQUINA.

El lenguaje propio del ordenador, basado en el sistema binario, o código máquina, resulta difícil de utilizar para las personas. El programador debe introducir todos y cada uno de los comandos y datos en forma binaria, y una operación sencilla como comparar el contenido de un registro con los datos situados en una ubicación del chip de memoria puede tener el siguiente formato: 11001010 00010111 11110101 00101011. La programación en lenguaje máquina es una tarea tan tediosa y consume tanto tiempo que muy raras veces lo que se ahorra en la ejecución del programa justifica los días o semanas que se han necesitado para escribir el mismo.

2.1.3 LENGUAJE ENSAMBLADOR.

Uno de los métodos inventados por los programadores para reducir y simplificar el proceso es la denominada programación con lenguaje ensamblador. Al asignar un código mnemotécnico (por lo general de tres letras) a cada comando en lenguaje máquina, es posible escribir y depurar o eliminar los errores lógicos y de datos en los programas escritos en lenguaje ensamblador, empleando para ello sólo una fracción del tiempo necesario para programar en lenguaje máquina. En el lenguaje ensamblador, cada comando mnemotécnico y sus operadores simbólicos equivalen a una instrucción de máquina. Un programa ensamblador traduce el

código fuente, una lista de códigos de operación mnemotécnicos y de operadores simbólicos, a código objeto (es decir, a lenguaje máquina) y, a continuación, ejecuta el programa.

Sin embargo, el lenguaje ensamblador puede utilizarse con un solo tipo de chip de CPU o microprocesador. Los programadores, que dedicaron tanto tiempo y esfuerzo al aprendizaje de la programación de un ordenador, se veían obligados a aprender un nuevo estilo de programación cada vez que trabajaban con otra máquina. Lo que se necesitaba era un método abreviado en el que un enunciado simbólico pudiera representar una secuencia de numerosas instrucciones en lenguaje máquina, y un método que permitiera que el mismo programa pudiera ejecutarse en varios tipos de máquinas. Estas necesidades llevaron al desarrollo de lenguajes de alto nivel.

2.1.4 LENGUAJES DE BAJO NIVEL.

Vistos a muy bajo nivel, los microprocesadores procesan exclusivamente señales electrónicas binarias. Dar una instrucción a un microprocesador supone en realidad enviar series de unos y ceros espaciadas en el tiempo de una forma determinada. Esta secuencia de señales se denomina código máquina. El código representa normalmente datos y números e instrucciones para manipularlos. Un modo más fácil de comprender el código máquina es dando a cada instrucción un mnemónico, como por ejemplo STORE, ADD o JUMP. Esta abstracción da como resultado el ensamblador, un lenguaje de muy bajo nivel que es específico de cada microprocesador.

Los lenguajes de bajo nivel permiten crear programas muy rápidos, pero que son a menudo difíciles de aprender. Más importante es el hecho de que los programas escritos en un bajo nivel son prácticamente específicos para cada

procesador. Si se quiere ejecutar el programa en otra máquina con otra tecnología, será necesario rescribir el programa desde el principio.

2.1.5 LENGUAJES DE ALTO NIVEL

Por lo general se piensa que los ordenadores son máquinas que realizan tareas de cálculos o procesamiento de textos. La descripción anterior es sólo una forma muy esquemática de ver una computadora. Hay un alto nivel de abstracción entre lo que se pide a la computadora y lo que realmente comprende. Existe también una relación compleja entre los lenguajes de alto nivel y el código máquina.

Los lenguajes de alto nivel son normalmente fáciles de aprender porque están formados por elementos de lenguajes naturales, como el inglés. En Basic, el lenguaje de alto nivel más conocido, los comandos como “IF CONTADOR = 10 THEN STOP” pueden utilizarse para pedir a la computadora que pare si CONTADOR es igual a 10. Por desgracia para muchas personas esta forma de trabajar es un poco frustrante, dado que a pesar de que las computadoras parecen comprender un lenguaje natural, lo hacen en realidad de una forma rígida y sistemática.

Los lenguajes de alto nivel suelen utilizar términos ingleses del tipo LIST, PRINT u OPEN como comandos que representan una secuencia de decenas o de centenas de instrucciones en lenguaje máquina. Los comandos se introducen desde el teclado, desde un programa residente en la memoria o desde un dispositivo de almacenamiento, y son interceptados por un programa que los traduce a instrucciones en lenguaje máquina.

Los programas traductores son de dos tipos: intérpretes y compiladores. Con un intérprete, los programas que repiten un ciclo para volver a ejecutar parte

de sus instrucciones, reinterpretan la misma instrucción cada vez que aparece. Por consiguiente, los programas interpretados se ejecutan con mucha mayor lentitud que los programas en lenguaje máquina. Por el contrario, los compiladores traducen un programa íntegro a lenguaje máquina antes de su ejecución, por lo cual se ejecutan con tanta rapidez como si hubiesen sido escritos directamente en lenguaje máquina.

Se considera que fue la estadounidense Grace Hopper quien implementó el primer lenguaje de ordenador orientado al uso comercial. Después de programar un ordenador experimental en la Universidad de Harvard, trabajó en los modelos UNIVAC I y UNIVAC II, desarrollando un lenguaje de alto nivel para uso comercial llamado FLOW-MATIC. Para facilitar el uso del ordenador en las aplicaciones científicas, IBM desarrolló un lenguaje que simplificaría el trabajo que implicaba el tratamiento de fórmulas matemáticas complejas. Iniciado en 1954 y terminado en 1957, el FORTRAN (acrónimo de Formula Translator) fue el primer lenguaje exhaustivo de alto nivel de uso generalizado.

En 1957 una asociación estadounidense, la Association for Computing Machinery comenzó a desarrollar un lenguaje universal que corrigiera algunos de los defectos del FORTRAN. Un año más tarde fue lanzado el ALGOL (acrónimo de Algorithmic Language), otro lenguaje de orientación científica. De gran difusión en Europa durante las décadas de 1960 y 1970, desde entonces ha sido sustituido por nuevos lenguajes, mientras que el FORTRAN continúa siendo utilizado debido a las gigantescas inversiones que se hicieron en los programas existentes. El COBOL (acrónimo de Common Business Oriented Language) es un lenguaje de programación para uso comercial y empresarial especializado en la organización de datos y manipulación de archivos, y hoy día está muy difundido en el mundo empresarial.

El lenguaje Basic (acrónimo de Código de Instrucciones Simbólicas de Uso General para Principiantes) fue desarrollado en el Dartmouth College a principios

de la década de 1960 y está dirigido a los usuarios de ordenador no profesionales. Este lenguaje se universalizó gracias a la popularización de los microordenadores en las décadas de 1970 y 1980. Calificado de lento, ineficaz y poco estético por sus detractores, Basic es sencillo de aprender y fácil de utilizar. Como muchos de los primeros microordenadores se vendieron con Basic incorporado en el hardware (en la memoria ROM), se generalizó el uso de este lenguaje.

Aunque existen centenares de lenguajes informáticos y de variantes, hay algunos dignos de mención, como el PASCAL, diseñado en un principio como herramienta de enseñanza, hoy es uno de los lenguajes de microordenador más populares; el Logo fue desarrollado para que los niños pudieran acceder al mundo de la informática; el C, un lenguaje de Bell Laboratories diseñado en la década de 1970, se utiliza ampliamente en el desarrollo de programas de sistemas, al igual que su sucesor, el C++. El LISP y el PROLOG han alcanzado amplia difusión en el campo de la inteligencia artificial.

2.1.6 INTÉRPRETES Y COMPILADORES.

La traducción de una serie de instrucciones en lenguaje ensamblador (el código fuente) a un código máquina (o código objeto) no es un proceso muy complicado y se realiza normalmente por un programa especial llamado compilador. La traducción de un código fuente de alto nivel a un código máquina también se realiza con un compilador, en este caso más complejo, o mediante un intérprete. Un compilador crea una lista de instrucciones de código máquina, el código objeto, basándose en un código fuente. El código objeto resultante es un programa rápido y listo para funcionar, pero que puede hacer que falle el ordenador si no está bien diseñado. Los intérpretes, por otro lado, son más lentos que los compiladores ya que no producen un código objeto, sino que recorren el código fuente una línea cada vez. Cada línea se traduce a código máquina y se ejecuta. Cuando la línea se lee por segunda vez, como en el caso de los

programas en que se reutilizan partes del código, debe compilarse de nuevo. Aunque este proceso es más lento, es menos susceptible de provocar fallos en la computadora.

2.2 TIPOS DE PROGRAMACIÓN.

Las instrucciones deben darse en un lenguaje de programación, es decir, en una determinada configuración de información digital binaria. En las primeras computadoras, la programación era una tarea difícil y laboriosa ya que los conmutadores ON-OFF de las válvulas de vacío debían configurarse a mano. Programar tareas tan sencillas como ordenar una lista de nombres requería varios días de trabajo de equipos de programadores. Desde entonces se han inventado varios lenguajes informáticos, algunos orientados hacia funciones específicas y otros centrados en la facilidad de uso.

2.2.1 PROGRAMACIÓN ESTRUCTURADA.

Programación estructurada, en informática, término general que se refiere a un tipo de programación que produce código con un flujo limpio, un diseño claro y un cierto grado de modularidad o de estructura jerárquica. Entre los beneficios de la programación estructurada se encuentran la facilidad de mantenimiento y la legibilidad por parte de otros programadores.

La programación estructurada se refiere tanto a la estrategia del programador, como al lenguaje utilizado. Así, para ella se usan lenguajes de programación que faciliten el diseño de aplicaciones con llamadas a procedimientos o funciones, como lo son típicamente Pascal y Ada, entre otros. En ellos el flujo de información es más fácilmente legible y no requieren de bifurcaciones basadas en llamadas a líneas concretas (uso de etiquetas al estilo Basic), sino, más bien, en saltos a áreas de código perfectamente diferenciadas.

La programación bien estructurada permite, de forma adicional, la reusabilidad del código, extrayendo módulos que pueden ser utilizados en otros programas, sin cambios en el código o con un mínimo de readaptaciones.

2.2.2 PROGRAMACIÓN LINEAL.

Programación lineal, técnica matemática y de investigación de operaciones que se utiliza en la planificación administrativa y económica para maximizar las funciones lineales de un gran número de variables sujetas a determinadas restricciones. El desarrollo de computadoras electrónicas y de técnicas de procesamiento de alta velocidad ha aportado recientemente muchos avances a la programación lineal, de forma que ahora esta técnica se utiliza extensamente en operaciones industriales y militares.

La programación lineal se utiliza básicamente para hallar un conjunto de valores, elegidos a partir de un conjunto de números dado, que maximizarán o minimizarán una forma polinómica dada. En el siguiente ejemplo se muestra un tipo particular de problema y un método para solucionarlo. Un fabricante produce dos variantes, V1 y V2, de un artículo que contiene piezas que se deben cortar, ensamblar y acabar. El fabricante sabe que puede vender tantos artículos como produzca. La variante V1 requiere 25 minutos de corte, 60 minutos de ensamblaje y 68 minutos de acabado, generando un beneficio de 30 dólares. La variante V2 requiere 75 minutos de corte, 60 minutos de ensamblaje y 34 minutos de acabado, generando 40 dólares de beneficio. Cada día se dispone de un máximo de 450 minutos de corte, 480 minutos de ensamblaje y 476 minutos de acabado. ¿Cuántos artículos de cada variante deben fabricarse diariamente para maximizar los beneficios?

Sean x e y los números de artículos de las variedades V1 y V2, respectivamente, que deben ser fabricados diariamente para maximizar los beneficios. Dado que x e y no pueden ser números negativos.

$$x \geq 0 \qquad (1)$$
$$y \geq 0 \qquad (2)$$

Los datos de corte, ensamblaje y acabado, determinan las siguientes igualdades y desigualdades:

$$25x + 75y \leq 450 \qquad (3)$$
$$60x + 60y \leq 480 \qquad (4)$$
$$68x + 34y \leq 476 \qquad (5)$$

En un gráfico, estas desigualdades representan áreas bajo líneas dadas.

El problema radica en hallar los valores de x e y que maximizarán el beneficio, si existen, siempre que cumplan las restricciones (1) a (5).

Para satisfacer las cinco condiciones, el punto que representa x e y debe hallarse en el límite o en el interior de la región convexa poligónica OABCD de la figura 1.

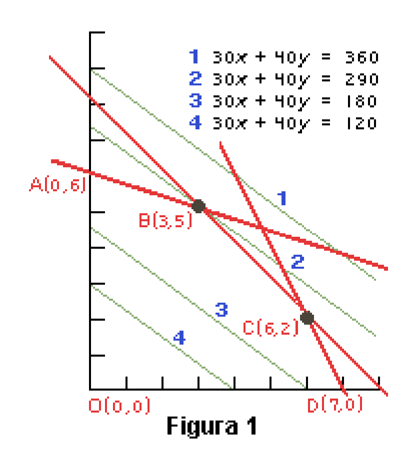


Figura 1

El beneficio será máximo eligiendo la línea definida por p =30 x + 40 y, donde p se encuentra en el máximo y sólo roza la región OABCD superior, es decir, la línea que atraviesa el vértice B (3,5). El fabricante ingresará los máximos

beneficios (290 dólares) produciendo 3 artículos de la variedad V1 y 5 artículos de la variedad V2 al día. Cualquier otra cantidad de ambas variantes, dadas las restricciones en cuanto al tiempo, reducirá el beneficio.

2.2.3 PROGRAMACIÓN ORIENTADA A OBJETOS (POO).

Programación orientada a objetos, en informática, un método de programación en el que un programa se contempla como un conjunto de objetos limitados que, a su vez, son colecciones independientes de estructuras de datos y rutinas que interactúan con otros objetos, o que dan acceso para modificar el contenido de un dato o propiedad del propio objeto. También se identifica con las siglas POO y OOP (del inglés Object Oriented Programming).

El lenguaje Simula (Simulation Language), desarrollado en 1964, partiendo del lenguaje ALGOL 60 (Algorithmic Language), fue el primero que se diseñó para facilitar la programación orientada a objetos.

El concepto clave en la programación orientada a objetos es la denominada clase (de objetos). Una clase define las estructuras de datos y rutinas de un objeto. A su vez, un objeto es una instancia de una clase, que se puede usar como una variable en un programa; es decir, la creación de una instancia de una clase se corresponde con la declaración de una variable en la programación tradicional, pero refiriéndose a objetos. En algunos lenguajes orientados a objetos, el objeto responde a mensajes, que son el principal medio de comunicación. En otros lenguajes orientados a objetos se conserva el mecanismo tradicional de llamadas a procedimientos o funciones (según los casos).

En cuanto a la funcionalidad y la metodología interna, la programación orientada a objetos se fundamenta en la encapsulación, la herencia y el poliformismo. La encapsulación significa que en una clase se declaran los tipos de

datos y el medio de manipularlos (sus métodos). La herencia supone crear clases derivadas de otras existentes, que heredan sus tipos y métodos y pueden contener otros nuevos. Si una nueva clase hereda propiedades de más de una antecesora, se denomina herencia múltiple; no todos los lenguajes orientados a objetos lo permiten. El poliformismo facilita la programación de funciones o procedimientos que ejecutarán acciones que dependerán de los objetos sobre los que se apliquen; por ejemplo, aumentar el tamaño de un objeto, independientemente de su forma.

En La actualidad, los lenguajes de programación orientada a objetos más utilizados son el lenguaje C++, JAVA, Smalltalk, Pascal orientado a objetos (Object Pascal) y otros, bien derivados, bien precursores de éstos.

2.2.4 CÓDIGO ASCII.

ASCII, acrónimo de American Standard Code for Information Interchange (Código Normalizado Americano para el Intercambio de Información). En computación, un esquema de codificación que asigna valores numéricos a las letras, números, signos de puntuación y algunos otros caracteres. Al normalizar los valores utilizados para dichos caracteres, ASCII permite que los ordenadores o computadoras y programas informáticos intercambien información.

ASCII incluye 256 códigos divididos en dos conjuntos, estándar y extendido, de 128 cada uno. Estos conjuntos representan todas las combinaciones posibles de 7 u 8 bits, siendo esta última el número de bits en un byte. El conjunto ASCII básico, o estándar, utiliza 7 bits para cada código, lo que da como resultado 128 códigos de caracteres desde 0 hasta 127 (00H hasta 7FH hexadecimal). El conjunto ASCII extendido utiliza 8 bits para cada código, dando como resultado 128 códigos adicionales, numerados desde el 128 hasta el 255 (80H hasta FFH extendido).

En el conjunto de caracteres ASCII básico, los primeros 32 valores están asignados a los códigos de control de comunicaciones y de impresora caracteres no imprimibles, como retroceso, retorno de carro y tabulación empleados para controlar la forma en que la información es transferida desde una computadora a otra o desde una computadora a una impresora. Los 96 códigos restantes se asignan a los signos de puntuación corrientes, a los dígitos del 0 al 9 y a las letras mayúsculas y minúsculas del alfabeto latino.

Los códigos de ASCII extendido, del 128 al 255, se asignan a conjuntos de caracteres que varían según los fabricantes de computadoras y programadores de software. Estos códigos no son intercambiables entre los diferentes programas y computadoras como los caracteres ASCII estándar. Por ejemplo, IBM utiliza un grupo de caracteres ASCII extendido que suele denominarse conjunto de caracteres IBM extendido para sus computadoras personales. Apple Computer utiliza un grupo similar, aunque diferente, de caracteres ASCII extendido para su línea de computadoras Macintosh. Por ello, mientras que el conjunto de caracteres ASCII estándar es universal en el hardware y el software de los microordenadores, los caracteres ASCII extendido pueden interpretarse correctamente sólo si un programa, computadora o impresora han sido diseñados para ello.

Las letras mayúsculas y los dígitos numéricos se representaran en el conjunto de caracteres ASCII con el conjunto de 36 patrones únicos de ocho bits. Los otros 92 patrones ASCII representan minúsculas, signos de puntuación y caracteres especiales.

2.2.5 PSEUDOCÓDIGO.

La forma de escribir estructuras es por medios simbólicos, usando notación de pseudocódigo. El pseudocódigo, requiere de símbolos privilegiados que ya tienen significado preciso y establecido de antemano. A tales indicadores se les conoce como palabras clave (keywords). Es necesario que exista una palabra

clave para la selección y otra para la insistencia condicional, así como para las instrucciones adicionales y otras estructuras de control.

En virtud de que las palabras clave son palabras que se hablan acerca de otras, adquieren la categoría de meta palabras, razón por la cual se deben distinguir de las que no lo son, y eso se logra subrayándolas, o escribiéndolas con otro tipo de letra.

2.3 DESARROLLO DE UN PROGRAMA.

Escribir programas para computadora es una actividad que requiere una buena cantidad de tiempo y esfuerzo menta. Armados ya que de la descripción teórico-conceptual sobre las computadoras y los algoritmos, se emprende el camino de volver a la realidad la construcción de pequeños modelos para representar descripciones de fenómenos o procesos del mundo real.

Esto implica una metodología científica, repetible y comprobable. Se hablará ahora del proceso mental asociado con la construcción de programas para la PC´s.

Las fases en la construcción de un programa son, en orden, las siguientes (aunque debe quedar claro que no hay límites tajantes entre el final de una y el inicio de otra):

0 Entender el problema o la situación.
1 Hacer el análisis (a veces este paso se denomina análisis de sistema).
2 Programar el modelo de solución propuesto.
3 Codificarlo.
4 Llevarlo a la computadora para su ejecución y ajuste.
5 Darle mantenimiento durante su tiempo de vida.

El paso cero parece banal pero deja de serlo cuando se piensa en la gran cantidad de proyectos de computación que se desarrollan (y a veces se terminan) sin haber comprendido cabalmente para qué eran, o cuál era el problema que supuestamente iban a resolver. Y si, además, se toma en cuenta que los sistemas de programación reales, a diferencia de las tareas y ejercicios de carácter didáctico o académico, suelen ser elaborados y complejos e implican la participación de varias personas (a veces decenas o cientos de ellas) durante largos períodos, se podrá comprender la importancia de entender con claridad el problema antes de abocar recursos a su solución.

2.3.1 ANÁLISIS DE SISTEMA.

Consiste en efectuar un análisis completo del problema o sistema existente, con la finalidad de proponer un modelo para su solución. Debe ser claro que este modelo no puede existir sin que se hayan especificado con claridad todos y cada uno de los componentes estructurales del sistema.

La estructura de un sistema es la forma en que están relacionados entre sí sus diversos componentes, de modo que resulta perfectamente posible tener sistemas distintos con componentes iguales. La diferencia estará en la forma de hacer corresponder unos con otros.

Para analizar se dispone de varios enfoques cualitativos, cuya finalidad consiste en proponer el lugar y la función de los componentes aislables del sistema del sistema (que pueden ser datos –objetos- o procedimientos), en términos tanto de los demás como de la función, ahora sí, que será desempeñada por el conjunto.

El análisis de sistemas es una actividad compleja y altamente dependiente de consideraciones humanas; por lo tanto, no ha sido aún comprendida en su

totalidad dentro de un esquema matemático. Esto significa que las experiencias previas en el análisis son factor primordial en el desarrollo de uno nuevo, y de que no existe, una forma “segura” de lograr un análisis correcto o productivo en primera instancia, sino que el proceso está sujeto a mejoras, que pueden ser producto de esquemas inductivos o de simples ensayos de prueba y error. Para sistemas complejos, existen elaboradas técnicas de ingeniería de software.

2.3.2 PROGRAMACIÓN.

Cuando se ha realizado el análisis, se debe escribir en pseudocódigo.

Las estructuras de control son las formas que existen para dirigir el flujo de acciones que el procesador efectuará sobre los datos que se manejen en un programa, mismos que están organizados de maneras diversas.

La definición mínima de un programa: un conjunto de declaraciones de estructuras de datos seguidas de un conjunto de proposiciones ejecutables (que incluye todos los componentes de las estructuras de control). Además, este programa o cadena de símbolos válidos cumple otra condición: está bien formado. Una cadena bien formada se arma siguiendo reglas sintácticas (en el sentido definido de los compiladores) de la gramática que produce el lenguaje empleado.

Como en esté caso no se está hablando de ningún lenguaje de programación en particular, entonces un programa bien formado cuando sea el producto de la aplicación de ciertas reglas de construcción primitivas.

2.3.3 CODIFICACIÓN.

Una vez terminada la programación se habrá producido una descripción del modelo propuesto, escrita en pseudocódigo. La razón de ser de este paso

disponer de un programa que pueda ser probado mentalmente para averiguar si es correcto el principio, y para determinar a qué grado considera todo el análisis hecho anteriormente. El proceso mediante el cual se llega a un programa esencialmente correcto recibe el nombre el nombre de refinamientos progresivos.

Sin embargo, un programa es pseudocódigo no es ejecutable en una computadora, por lo que se requiere refinarlo más. El objetivo de estos refinamientos consiste en acercar el programa de pseudocódigo a un lenguaje de programación en particular.

2.3.4 EJECUCIÓN Y AJUSTE.

Cuando se tiene el programa codificado y compilado llega el momento de ejecutarlo y probarlo “sobre la marcha”; es decir, hacer que la computadora lo ejecute para evaluar los resultados.

Una nociva práctica usual –que tiende a desaparecer- es dedicar poco tiempo a las etapas de análisis, programación y enfocar la atención y los recursos a la codificación, razón por la cual la ejecución estará siempre plagada de errores. Existen dos tipos de fallas: de sintaxis y de lógica de programación. Los primeros son relativamente triviales, mientras que los segundos son causantes de frecuentes retrasos que sufren los proyectos de programación en todos los niveles de complejidad. La concepción de la prueba de un programa se ha desplazado de la ejecución de la programación en pseudocódigo.

2.3.5 MANTENIMIENTO.

Se debe considerar el hecho del mantenimiento para asegurar que el modelo ya sistematizado evolucione a un ritmo parecido al de la realidad que está siendo simulada. Tal vez llegue el momento en el que la realidad simulada por el

sistema haya cambiado cualitativamente, en cuyo caso se habla del término de la vida útil del sistema.

Hay que ser capaces de hacer alteraciones no estructurales al sistema con costo mínimo en recurso de análisis y programación, lo cual está asegurando si el sistema se ha construido de manera modular, y si dispone de la documentación adecuada que lo describa tanto en su diseño como en su uso. Si un sistema sólo es comprensible por sus creadores es un mal sistema.

Esta falta de flexibilidad, además, resulta imposible de tolerar para el caso de sistemas grandes, creados por cientos de ingenieros en sistemas y programadores (el sistema operativo de la serie 360 de IBM requirió cerca de 5000 hombres para su desarrollo, y los actuales sistemas operativos de computadoras personales constan de varios millones de renglones fuente). No puede ser que un sistema de tal magnitud no tenga previstos cambios y adaptaciones constantes. Entonces existe el compromiso de que cualquier programa se tiene que realizar claro y flexible.

2.4 APLICACIÓN DE LOS LENGUAJES DE PROGRAMACIÓN.

Los lenguajes de programación permiten comunicarse con los ordenadores o computadoras. Una vez identificada una tarea, el programador debe traducirla o codificarla a una lista de instrucciones que la computadora entienda. Un programa informático para determinada tarea puede escribirse en varios lenguajes. Según la función, el programador puede optar por el lenguaje que implique el programa menos complicado. También es importante que el programador elija el lenguaje más flexible y más ampliamente compatible para el caso de que el programa tenga varias aplicaciones. Los ejemplos que se ven en la ilustración son programas escritos para calcular el promedio de una serie de números. C y BASIC son los lenguajes de computadora más utilizados. En el recuadro inferior de la ilustración

se muestra cómo una computadora procesará y ejecutará los comandos de los programas.

2.4.1 AUTOMATIZACIÓN.

Automatización, sistema de fabricación diseñado con el fin de usar la capacidad de las máquinas para llevar a cabo determinadas tareas anteriormente efectuadas por seres humanos, y para controlar la secuencia de las operaciones sin intervención humana. El término automatización también se ha utilizado para describir sistemas no destinados a la fabricación en los que dispositivos ***programados*** o automáticos pueden funcionar de forma independiente o semi independiente del control humano. En comunicaciones, aviación y astronáutica, dispositivos como los equipos automáticos de conmutación telefónica, los pilotos automáticos y los sistemas automatizados de guía y control se utilizan para efectuar diversas tareas con más rapidez o mejor de lo que podría hacerlo un ser humano.

La fabricación automatizada surgió de la íntima relación entre fuerzas económicas e innovaciones técnicas como la división del trabajo, la transferencia de energía y la mecanización de las fábricas, y el desarrollo de las máquinas de transferencia y sistemas de realimentación, como se explica a continuación.

La división del trabajo (esto es, la reducción de un proceso de fabricación o de prestación de servicios a sus fases independientes más pequeñas) se desarrolló en la segunda mitad del siglo XVIII, y fue analizada por primera vez por el economista británico Adam Smith en su libro Investigación sobre la naturaleza y causas de la riqueza de las naciones (1776). En la fabricación, la división del trabajo permitió incrementar la producción y reducir el nivel de especialización de los obreros.

La mecanización fue la siguiente etapa necesaria para la evolución hacia la automatización. La simplificación del trabajo permitida por la división del trabajo también posibilitó el diseño y construcción de máquinas que reproducían los movimientos del trabajador. A medida que evolucionó la tecnología de transferencia de energía, estas máquinas especializadas se motorizaron, aumentando así su eficacia productiva. El desarrollo de la tecnología energética también dio lugar al surgimiento del sistema fabril de producción, ya que todos los trabajadores y máquinas debían estar situados junto a la fuente de energía.

La máquina de transferencia es un dispositivo utilizado para mover la pieza que se está trabajando desde una máquina herramienta especializada hasta otra, colocándola de forma adecuada para la siguiente operación de maquinado. Los robots industriales, diseñados en un principio para realizar tareas sencillas en entornos peligrosos para los trabajadores, son hoy extremadamente hábiles y se utilizan para trasladar, manipular y situar piezas ligeras y pesadas, realizando así todas las funciones de una máquina de transferencia. En realidad, se trata de varias máquinas separadas que están integradas en lo que a simple vista podría considerarse una sola.

En la década de 1920 la industria del automóvil combinó estos conceptos en un sistema de producción integrado. El objetivo de este sistema de línea de montaje era abaratar los precios. A pesar de los avances más recientes, éste es el sistema de producción con el que la mayoría de la gente asocia el término automatización.

Un elemento esencial de todos los mecanismos de control automático es el principio de realimentación, que permite al diseñador dotar a una máquina de capacidad de auto corrección. Un ciclo o bucle de realimentación es un dispositivo mecánico, neumático o electrónico que detecta una magnitud física como una temperatura, un tamaño o una velocidad, la compara con una norma

preestablecida, y realiza aquella acción preprogramada necesaria para mantener la cantidad medida dentro de los límites de la norma aceptable.

El advenimiento del ordenador o computadora ha facilitado enormemente el uso de ciclos de realimentación en los procesos de fabricación. En combinación, las computadoras y los ciclos de realimentación han permitido el desarrollo de máquinas controladas numéricamente (cuyos movimientos están controlados por papel perforado o cintas magnéticas) y centros de maquinado (máquinas herramientas que pueden realizar varias operaciones de maquinado diferentes).

La aparición de las combinaciones de microprocesadores y computadoras ha posibilitado el desarrollo de la tecnología de diseño y fabricación asistidos por computadora (CAD/CAM). Empleando estos sistemas, el diseñador traza el plano de una pieza e indica sus dimensiones con la ayuda de un ratón o mouse, un lápiz óptico u otro dispositivo de introducción de datos. Una vez que el boceto ha sido terminado, la computadora genera automáticamente las instrucciones que dirigirán el centro de maquinado para elaborar dicha pieza.

Otro avance que ha permitido ampliar el uso de la automatización es el de los sistemas de fabricación flexibles (FMS). Los FMS han llevado la automatización a las empresas cuyos bajos volúmenes de producción no justificaban una automatización plena. Se emplea una computadora para supervisar y dirigir todo el funcionamiento de la fábrica, desde la programación de cada fase de la producción hasta el seguimiento de los niveles de inventario y de utilización de herramientas.

Asimismo, aparte de la fabricación, la automatización ha influido enormemente sobre otras áreas de la economía. Se utilizan computadoras pequeñas en sistemas denominados procesadores de textos, que se están convirtiendo en la norma de la oficina moderna. Esta tecnología combina una pequeña computadora con una pantalla de monitor de rayos catódicos, un teclado

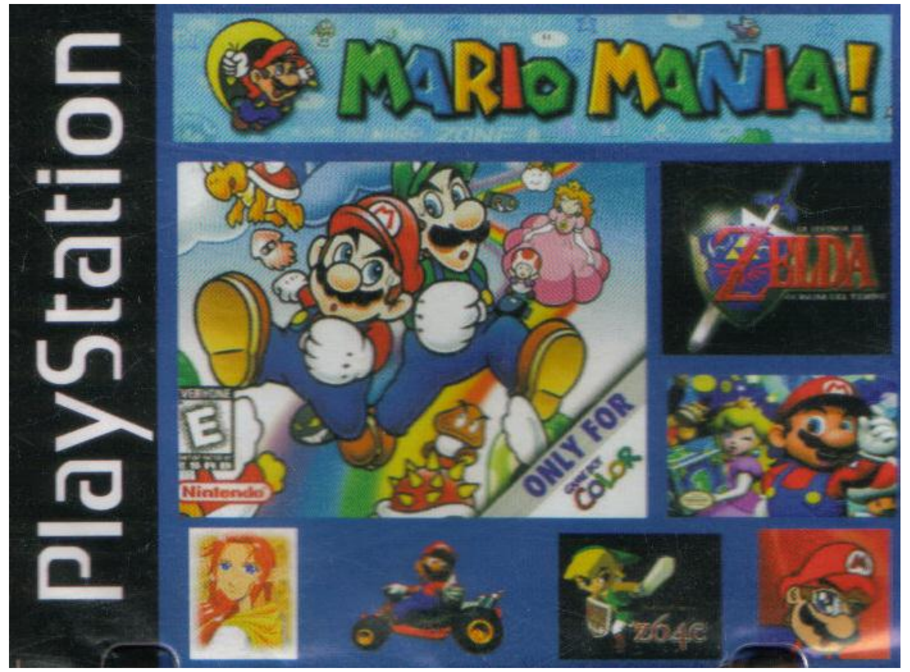
de máquina de escribir y una impresora. Se utiliza para editar texto, preparar cartas modelos personalizadas para su destinatario y gestionar listas de correo y otros datos. El sistema es capaz de realizar muchas otras tareas que han incrementado la productividad de la oficina.

Las aplicaciones de la programación no es solamente la automatización, sino un sin fin de tareas ya que se puede estructurar un programa para una calculadora o uno que lleve como tarea controlar las comunicaciones, navegar en Internet, controladores de dispositivos del CPU, un video juego, un procesador de texto, un reproductor multimedia, controlar misiles dirigidos por computadora, etc.

La programación lleva como tarea la que el programador desee darle a un ordenador.

Como conclusión de esté capítulo se ha explicado a grandes rasgos como funciona la computadora a través del software, que se tiene codificar, compilar, y comprobar su funcionalidad independientemente del tipo de lenguaje en el que se halla desarrollado, con sus palabras reservadas que existe para cada uno de ellos, se reviso los niveles de cada lenguaje de programación, así como también los pasos para la estructuración de un programa de computador. De igual forma se analizo la automatización como una de las principales aplicaciones de la programación en la actualidad.

CAPÍTULO III
PROGRAMACIÓN APLICADA EN VIDEO JUEGOS



OBJETIVO.

Se presentará la historia del video juego para conocer las aplicaciones de la programación en el videogame; se analizarán los pasos que se deben realizar en la elaboración del software de entretenimiento hasta su demostración.

El video juego como tal es dueño de gran curiosidad de personas de todas las edades, la mayoría de la gente a podido ver o jugar alguna vez uno de estos juegos, pero pocos se preguntan que hay detrás de ellos, por está razón se hablara un poco de la historia de ellos, y la estructura que necesita un video juego, así como todas las funciones de las que se requiere para su elaboración. Se expondrá dos tipos de posibilidades para constituir un video juego para aquella persona que se interesa en crear un juego de estos.

3.1 HISTORIA DEL VIDEO JUEGO.

La primera consola casera apareció en Estados Unidos en 1972 y, desde entonces no ha dejado de evolucionar.

El verdadero boom de los video juegos surgió cuando estos comienzan a entrar a los hogares de todo el mundo. Paralelamente al éxito de los primeros arcades como pong o Space Invaders, nacía el afán de la industria electrónica por llevar esos éxitos al ámbito doméstico. Apareció entonces la consola Odyssey de Magnavox, con una versión del pong llamado ping pong, tan simple que ni siquiera tenia marcador. Sin duda, fue la compañía Atari quien triunfó con una versión del pong más perfecta.

Realmente, estas cajas electrónicas no se podrían denominar video consolas, sino video juegos en sí mismos, ya que sólo disponían de un juego grabado en un chip. Hasta la llegada de los juegos intercambiables contenidos en cartuchos no surgió el significado de las video consolas. Este sistema apareció por primera vez con la consola Fairchild Channer F, pero fue de nuevo el Atari quien revolucionó el formato del cartucho con su modelo 2600 Video Computer System. Sus juegos no tenían tanta calidad como los de Intellivision o Colevisión; sin embargo, permitían la posibilidad al usuario elegir entre distintas variaciones del juego, alargando su vida y su jugabilidad.

Por aquel entonces, aparecían en los hogares los primeros microordenadores personales de 8 bits –como el ZX –Spectrum o Commodore VIC20- más potentes para jugar. No obstante, la gente seguía prefiriendo las consolas por su sencillo uso y la cantidad de cartuchos que había. Además, triunfaban los arcades de recreativas como Space Invaders, Pacman, Galaxian, Missile Command o Tank y las consolas brindaban la oportunidad de jugar a buenas versiones de estos juegos cómodamente ante el televisor.

Fue Space Invaders el primer juego de recreativas que se licenció para la introducción en un cartucho de Atari 2600, la cual no duró mucho en obtener toda exclusividad de los demás éxitos.

A finales de los 70 y principio de los 80, Atari era el líder indiscutible y muchas empresas fabricaban cartuchos de juego para su consola. A partir de ahí empezó el declive de la empresa debido a la gestión de los negocios y el gran número de juegos malos que no se vendían. Aunque Atari fabricó algunos sistemas más, con juegos mejores, fue la compañía nipona Nintendo la que cogió el testigo del éxito. Lanzó al mercado la NES (Nintendo Entertaiment System), la primera consola de video juegos con tecnología de 8 bits, con unas posibilidades para gráficos y sonidos comparables con los microordenadores personales de la época.

Su verdadero éxito llegó con el lanzamiento de la saga de Mario Bros, juego de plataformas de concepción única con un gran colorido y música, con una jugabilidad increíble, ideal para toda la familia. En Estados Unidos causó furor e incluso se organizaban concursos para jugadores. Fue tal su fama que Mario fue uno de los primeros personajes en llegar al cine y a la televisión. Para acompañar el éxito de Mario apareció en escena Zelda, creación también del señor Miyamoto.

No tardó mucho en aparecer una gran competidora también procedente de Japón, Sega. Llegó a occidente y se unió al mundo de los 8 bits con su consola Master System y su personaje bandera Sonic.

La saga de Sonic compitió, con su opuesto Mario Bros. Sin embargo, Sonic descubrió el poder de los juegos de plataformas con scroll parallax para consolas de 8 y 16 bits, con una velocidad y suavidad de movimiento increíbles para la época.

Técnicamente Sega era mejor que Nintendo, y su juego Sonic poseía una endiablada velocidad, mejor sonido y un diseño de niveles asombroso.

La batalla estaba servida entre las dos compañías, que propiciaron un resurgimiento del sector lúdico para el hogar. Surgieron nuevos modelos con potencia de 16 bits, como la Súper NES de Nintendo y la Sega Mega Drive. En esta ocasión, Nintendo superó a Sega. La Súper NES proporcionaba unos gráficos más llamativos y sonidos más claros.

Nintendo pasó directamente a los 64 bits, comercializando la consola Nintendo 64 para competir con Sega Saturn. Este avance de tecnología acompañaba a juegos cada vez más vistosos, lo que daba lugar a licencias de las recreativas del momento sin apenas perdidas de calidad.

No podemos olvidar equipos como la prestigiosa Neo-Geo de SNK, que sin lugar a dudas tenía mejores prestaciones que sus homólogas de Nintendo o Sega, o la Neo-Geo CD con juegos en compact disc. Sin embargo, su elevado precio (debido a que sus juegos eran los mismos que la compañía programaba para las recreativas) fue el motivo de su escaso éxito comercial.

En la época de los 64 bits, surgió la poderosa Sony, e introdujo al reinado a las consolas en los hogares con el Play Station (PSX). Sin duda, la consola que cuenta con el mayor número de juegos de la historia

Paralelamente el éxito de las video consolas, los juegos también se disfrutaban en otro tipo de aparatos.

Sony Computer Entertaiment of América (SCEA) dio ha conocer la bestial consola que salio el 4 de marzo de 2000 en Japón. El Play Station2 (PS2) la cual incluye lo siguiente: CPU de 128 bits, un procesador de 300 Mhz, un control análogo, tarjeta de memoria de 8 Mb, múltiples puertos de entrada para un MODEM y poder jugar en línea con video jugadores del resto del mundo, posee un drive de CD-ROM de 24X/ DVD de 4X: compatible con CD´s de música, películas y juegos DVD, además de ser compatible con los juegos del Play Station (PSX).

La respuesta de Mario y compañía a la amenaza de SCEA se llama Nintendo Game Cube la cual se forjo de una alianza con IBM y Panasonic; el primero ofreció el chip Gekko, una maravilla que corre los juegos a 400 Mhz. Panasonic por su parte provee los DVDs, está consola se lanzo al mercado desde 2001.

La epopeya de Microsoft en el mágico mundo del video juego con el Xbox el cual posee un poder gráfico de 128 bits, una velocidad de 733 Mhz, juegos en formato de CD-ROM y DVD, Microsoft lanzo al mercado la máquina más poderosa del mercado desde 2001: hasta tres veces más potente que el PS2. La cual incluye un disco duro de 8 Gb, además se incluye el servicio de Xbox Live que está muy bien diseñado para competir desde tu hogar con jugadores del resto del mundo.

El video juego y la obsesión por estar jugando llego a la creación de consolas portátiles entre las primeras y las más fuertes en su lanzamiento fue el

Game boy (GB) con una resolución de 8 bits el cual logro evolucionar en 1995 (GBC) para que sus juegos tuvieran 52 colores simultáneos, en un formato de cartuchos muy fácil de llevar de un lado a otro. En 2001 el último baluarte del juego en 2D es también el único sistema de la lista que puedes jugar en el retrete, además de llevar los éxitos en 2D del Game boy Advance (GBA) sin olvidar las translaciones de el Súper NES.

La filosofía del video juego con consolas portátiles en nuestro país es muy diferente a otras partes del mundo, ya que en los últimos cinco años es muy poco recurrente ver en la calle personas con un GB/GBC/GBA, esto no significa que estás consolas portátiles no se vendan en México, pero si existe la suposición de que la gente tiene miedo de de sacarlos a la calle. Así que esté tipo de consolas están confinadas a la casa y en el mejor de los casos, en el trabajo y los vehículos. O probablemente también nos da vergüenza que nos vean en la calle pasándola a todo dar.

En los próximos dos años tendrán llegada tres nuevos sistemas del actual monarca Sony, el creciente y optimista Microsoft y el viejo lobo de mar Nintendo.

	SONY.	**MICROSOFT.**	**NINTENDO.**
Nombre(s) clave.	Next-generation Play Station, Play Station 3.	Xenon, Xbox 2, Xbox next.	Revolution.
Fecha tentativa de revelación.	Marzo de 2005.	Marzo-mayo de 2005.	Mayo de 2005.
Fecha tentativa de lanzamiento.	Finales de 2006.	Finales de 2005.	Finales de 2006.
Alianza(s) tecnológica(s).	IBM y Toshiba (CPU Nvidia) y unidad de procesos gráficos (GPU)	IBM (CPU), ATI (GPU).	IBM (CPU) y ATI (GPU).

¿En línea?	Probablemente. El juego en línea será parte de las consolas futuras.	Se esperan novedades en Xbox Live.	Probablemente.
¿Compatibilidad retroactiva?	Muy posible.	Es difícil ya que hardware será diferente al del Xbox.	Podría ser la primera vez.
¿Disco duro?	No.	Si.	No.

3.1.1 EL NUEVO ENTRETENIMIENTO CASERO.

Videojuegos, juegos electrónicos cuyo desarrollo tiene lugar en la pantalla de un ordenador (computadora) o de una televisión, y en los que el programa va grabado en un disquete informático, un disco compacto (o CD) o un cartucho especial para juegos. Algunos son de bolsillo. La enorme popularidad alcanzada por estos juegos a finales de la década de 1970 ha dado origen a una importante industria.

Los jugadores emplean una serie de teclas o botones y una palanca, también llamada joystick. La partida se juega entre una persona y la máquina, o entre dos o más personas que compiten con la máquina o entre sí. Los más populares emplean sonidos reales y colores, además de rápidos efectos visuales. Los juegos deportivos, como el fútbol, el baloncesto o el jockey sobre hielo, adquirieron especial popularidad a finales de la década de 1980, cuando determinados equipos profesionales prestaron su nombre a estas versiones en vídeo de su deporte.

Los establecimientos dedicados a la venta exclusiva de videojuegos son ya clásicos en la industria del ocio del mundo entero. Los sistemas de 16 bits, introducidos por las empresas japonesas a comienzos de la década de 1990,

mejoraron notablemente la calidad de los gráficos de los videojuegos destinados al uso doméstico. El CD-ROM (un sistema de memoria con enorme capacidad de almacenamiento) ofrece hoy una amplia variedad de juegos a precios muy competitivos. En el futuro, el uso de la televisión de alta definición y de las líneas de telecomunicaciones para la transmisión de este tipo de juegos contribuirá a aumentar aún más su realismo.

Las críticas a los videojuegos parten de los efectos negativos que el hecho de pasar demasiado tiempo ante la pantalla e inhibirse por completo en un mundo de fantasía puede tener en el desarrollo emocional de los niños. También se ha comprobado que la rapidez con que se mueven los gráficos puede provocar ataques en las personas que padecen diversos tipos de epilepsia.

Los defensores de los videojuegos afirman que estos juegos enseñan a resolver problemas técnicos, estimulan la habilidad de los jugadores y familiarizan a los niños con el uso de los equipos informáticos. Incluso sostienen que mejoran la comunicación, cuando se juega en familia. Los videojuegos se emplean también como entretenimiento en clínicas y hospitales, así como en ciertas terapias de rehabilitación.

La industria del videojuego surgió con fuerza en Estados Unidos, Europa y Australia con la llegada del tenis de mesa y otros juegos muy sencillos a finales de la década de 1970 y se extendió rápidamente por todo el mundo. Tras conocer un crecimiento espectacular a lo largo de la década de 1980, la industria japonesa (especialmente la Nintendo Corporation) se lanzó de lleno a perfeccionar y desarrollar la tecnología del juego, introduciendo juegos tan populares como el Súper Mario. Las empresas afincadas en Japón, como Nintendo y Sega, continúan dominando el mercado mundial.

Desde 1993 estas dos compañías están realizando esfuerzos para controlar y establecer el contenido de los juegos. La iniciativa responde a las críticas,

especialmente de los padres, preocupados por la intensificación de la violencia y la introducción de temas para adultos en los juegos infantiles.

Independientemente de lo que diga la gente sobre el uso que se hace de las computadoras caseras, los estudiantes indican que sobre todo se emplean para jugar. Los juegos por computadora y las máquinas de video juegos (que no son más que computadores de propósito especial) representan una industria gigantesca que probablemente seguirá evolucionando con rapidez en los próximos años.

Es probable que los principales cambios en los juegos electrónicos se den cuando las tecnologías de computación y comunicación converjan en la industria del entretenimiento casero. Cuando esto suceda se desvanecerá la línea que separa a los programas de televisión de los juegos de computadora. Hace pocos años, las tiendas de software ofrecían diversos juegos de ficción interactiva: historias con primitivas interfaces en lenguaje natural con las que los jugadores tenían cierto control sobre la trama. Estos programas sin gráficos y poco inteligentes han sido desplazados del mercado por películas interactivas: programas animados en los cuales los espectadores controlan uno o más de los personajes.

3.1.2 SOFTWARE DE JUEGOS.

Existen muchos tipos de juegos disponibles, en su mayoría son simuladores por computadora. Pueden simular juegos de mesa, de acción, de cartas, eventos deportivos, batallas intergalácticas, peleas callejeras, compras corporativas o alguna otra cosa, real o imaginaria. Hay juegos para todas las edades y niveles de habilidad. Muchos requieren estrategias y la capacidad de resolver acertijos; otros dependen exclusivamente de la coordinación ojo-mano. Un buen número de los juegos más populares requieren un poco de cada cosa. Con sus deslumbrantes

gráficos, su sonido digitalizado y elaborados efectos, los juegos por computador representan la vanguardia en software. Pero dentro de unos cuantos años estos juegos parecerán tan anticuados como se ven ahora esos primeros juegos de pong.

Se pueden adquirir juegos en una tienda de computación. Existen además otros disponibles en Internet. Algunos juegos son muy costosos, pero muchos fabricantes ofrecen una versión de prueba sin costo alguno en la World Wide Web (www), de manera que pueda probarlo antes de comprarlo.

Muchos juegos están diseñados para permitirles a varias personas competir entre ellos en la Internet o una red.

Existen servicios muchos servicios de juegos en línea disponibles para permitirle conectarse fácilmente a otras personas en la Internet para jugar. Cuando juega con otras personas en la Internet, cada una tiene por lo general su propia copia.

Un controlador de juegos, como un joystick, o almohadillas de juegos, le permiten interactuar con un juego. Algunos controladores muy conocidos son con el Microsoft SideWinder Gravis Game Pad. Algunos juegos requieren un tipo específico de controlador.

Muchos juegos están diseñados con gráficos 3D y corren menores en una computadora con una tarjeta de gráficos 3D, la cual es una placa de circuito que traduce instrucciones complejas desde la computadora al monitor en una forma que pueda comprender. Algunas tarjetas gráficas 3D incluyen Matrox Millenium, ATI Radeon y 3dfx Voodoo. Algunos juegos requieren un tipo específico de tarjeta de gráficos 3D. Antes de comprar un juego revise cuál tipo de tarjeta requiere.

3.1.3 CUANDO LOS VIDEOJUEGOS DEJAN DE SER UN JUEGO.

A veces nos olvidamos de que detrás de un videojuego hay enormes intereses financieros, y eso no es un juego. Tan sólo en 2003 las ventas de juegos de consolas en Estados Unidos sumaron 5.8 millonardos (un millonardo son mil millones) de dólares (186.4 millones de juegos vendidos), en tanto que los juegos de PC representaron 1.2 millonardos de dólares (52.8 millones de juegos vendidos). En el 2002 las ventas fueron de $6.9 millonardos de dólares, de los cuales los juegos de consola significaron 5.5 millonardos y los juegos de PC contribuyeron con 1.4 millonardos de dólares. Algo importante es que las cifras anteriores (divulgadas por Entertainment Software Associaton) no sólo son de software. En 2003 nueve juegos de consolas vendieron más de un millón de unidades cada uno. 39 juegos de consolas vendieron más de 500,000 unidades y 83 excedieron la barrera de las 250,000 unidades. En 2002 las cifras equivalentes fueron de 6, 33 y 73 respectivamente.

"Para 2008, 40.2 millones de jugadores en todo el mundo se registrarán en línea con consolas de videojuegos", afirma la firma de investigación de mercado DFC Intelligence. "Es innegable que éste es el siguiente nivel en juegos", dijo el analista de NPD Richard Ow.

En base a los reportes de los años anteriores, la Entertainment Software Associaton (ESA), asociación estadounidense que representa a los editores de programas de videojuegos y de cómputo, calculan que las ventas combinadas de videojuegos (consolas + PC) sobrepasarán los 7 millardos de dólares por primera vez en la historia, y que un número sin precedente de videojuegos de consolas venderán entre 500,000 y un millón de unidades al finalizar 2004.

Se calcula que en E.U.A. hay más de 145 millones de video jugadores, es decir que el 50% de estadounidenses de seis o más años de edad juegan. La

edad promedio de un video jugador actual es de 29 años, y el jugador promedio se la pasa 6.5 horas semanales jugando.

"La gente está gastando más y más de su tiempo jugando y dicho tiempo se está restando de ir al cine, ver la televisión y leer los periódicos", dijo Charlene Li, principal analista de deportivos, medios y mercadotecnia de Forrester Research.

Por último un número final. En 2002 la recaudación en taquilla de los cines a nivel mundial fue de $20.4 millardos de dólares, y la de los videojuegos en todo el mundo fue de 30 millardos.

3.2 ZONA DE DESARROLLO.

El nacimiento de un video juego empieza por su concepción. Todo comienza con la idea, la ilusión de hacer realidad el fruto que la imaginación va alimentando.

Hay que tener en cuenta que una cosa es ver el juego en la mente y otra muy distinta cómo será en la realidad. Se cae en el tópico de diseñar ideas que van más allá de las posibilidades técnicas que se poseen. Ocurre entonces que el resultado final no corresponde con el diseño original del juego y se tiene que realizar cambios sobre la marcha, retrasando el desarrollo.

Para empezar el desarrollo de un video juego se debe estudiar la forma de llevar nuestra flamante idea al mundo real, de un modo práctico, factible y siempre dentro de nuestra capacidad de desarrollo.

Cuando tenemos la idea para un juego, lo primero que se debe pensar es de qué tipo será: acción, deportivo, aventura gráfica, puzzles, etc. Después, si será en 2D o 3D, y, por último, se buscara originalidad, no cabe duda de que la

experiencia personal de cada uno como jugador aflora irremediablemente, y nuestras ideas pasan por los juegos que marcaron estilo. Entonces al crear un proyecto y esté se parece a un video juego al que se ha jugado, o que se ha observado en Internet o revista especializada se pierde la idea de singularidad.

Hoy en día, a pesar de la cantidad de nuevos desarrollos en el sector, no abundan mucho las ideas nuevas. Casi todos los juegos, aparentemente, son parecidos y tienen multitud de aspectos en común, sin embargo, siempre existe algún factor que los hace distintos unos de otros: nuevas armas, una apariencia gráfica diferente, diseño de la interfaz de uso, un sistema de audio más realista o desarrollo de la jugabilidad poco común. Y es, en definitiva, la calidad y diseño de todos estos aspectos lo que hace que un video juego sea mejor o peor que otro y, en contadas ocasiones, que pase a crear un nuevo estilo.

3.2.1 LA IDEA DEL JUEGO.

La intención no es crear un estilo nuevo o marcar historia, simplemente se pretende utilizar una idea ya existente, para facilitar un método pedagógico práctico y cómodo que ayude a la comprensión del complejo arte de desarrollar videojuegos.

Partiremos de la premisa de un juego de combate en el que el jugador se enfrenta a uno o a varios contrincantes. Estos adversarios pueden ser manejados por el ordenador o pueden ser otros jugadores conectados a través de Internet o la red local.

El juego transcurre en una gran superficie formada por montes, ríos, bosques y construcciones. En ella se encuentran gusanos subterráneos letales y plantas carnívoras, entre otros peligros. Los jugadores disponen de una nave de combate con posibilidad de utilizar distintos tipos de armas y escudo de energía

para la protección. Se desplaza por el terreno flotando a cierta distancia del suelo, pudiendo acelerar y frenar a voluntad pero nunca parar del todo.

Asimismo, pueden sumar puntos por medio de bonos esparcidos por la superficie o aniquilando enemigos, animales o plantas. Vencerá el último jugador que quede vivo.

Estos terrenos no forman parte de ningún mapa o región sino que se encuentran en el interior de enormes urnas sobre pedestales, la “zona de luchadores” (zone of fighters).

3.2.2 LA HISTORIA.

Como cada año, en el sistema Hooman, se celebra el mayor acontecimiento de la galaxia, la prueba de los campeones, el desafío definitivo, el combate mortal en grandes terrarios cargados de peligros llamados zone of fighters.

Los participantes, venidos de todos los rincones, lucharán entre sí en bio-naves de combate diseñadas para ser guiadas con la mente. El reto, sobrevivir. Vencerá el último que quede con vida. No hay piedad para los cobardes.

Ya teniendo a grandes rasgos la idea del video juego. Y para complementarlo, se tendrá que definir algunos aspectos más.

3.2.3 TIPO DE JUEGO.

Se debe decidir si el juego será en 2D o 3D y a qué estilo pertenecerá. Según el planteamiento de nuestra idea, la forma más real y espectacular de llevarla a cabo sería en un entorno 3D real. En definitiva, zone of fighters será claramente un juego de acción 3D.

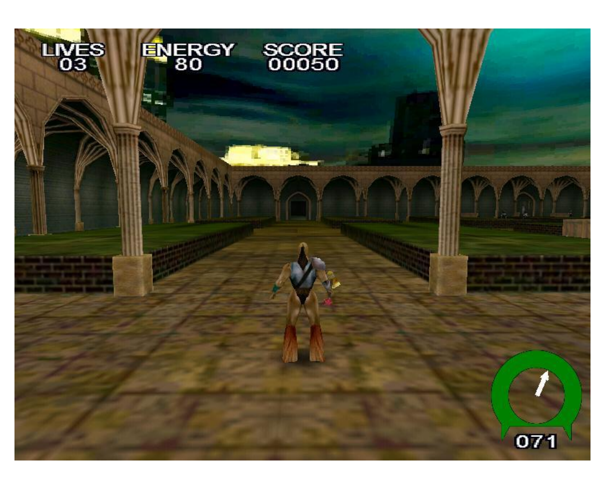
Actualmente, multitud de juegos suelen mezclar los géneros. Así, es fácil encontrarlos de acción-aventuras, acción-puzzles o acción-velocidad; lo que diferencia a cada uno es el tipo de desarrollo de la acción.

3.2.4 SISTEMA DE VISUALIZACIÓN.

Representar la acción de un juego por medio de un sistema de visualización en tres dimensiones ofrece multitud de ventajas, sobre todo al jugador, pero también ayuda a conseguir un mayor realismo. En esté sistema, el jugador pasa a ver la acción como si observara a través de una cámara situada en distintos puntos. Desaparece pues la limitación que existe en un sistema de dos dimensiones, donde la acción se representa desde una sola vista y los gráficos siempre con el mismo ángulo: de un lado u otro, de frente o espalda, desde arriba o abajo. Con una dimensión más, es posible ver el gráfico y el entorno desde todos sus ángulos y en tiempo real.

En las ilustraciones se exponen dos ejemplos de de posición de cámara. En la foto de la izquierda se muestra un juego con perspectiva en tercera persona y en la imagen de la derecha con perspectiva en primera persona (FPS).





Cambiando la posición de la cámara o la vista del protagonista principal se deberá modificar también el tamaño y posición de todos los gráficos. Estos

cambios hacen la distinción entre los juegos 3D. Por ejemplo, un FPS (First Person Shooter), o juego de primera persona, tendría situada la cámara en un plano subjetivo; es decir, el jugador nunca ve a su protagonista porque la cámara nos muestra lo que éste ve. Cuando se habla de un juego en tercera persona, la cámara está situada por encima y a la espalda del protagonista, como si una tercera persona estuviera viendo la acción. En ocasiones, los juegos en 3D se muestran con perspectivas utilizadas en juegos 2D, como la cenital (vista desde arriba) o la isométrica, pero con la particularidad de que, al ser 3D real, es posible girar la cámara alrededor del entorno.

En zone of fighters se puede utilizar la posibilidad de cambiar la posición de la cámara a nuestro antojo; de está forma obtendremos varias vistas del entorno. Daremos así la oportunidad al jugador de ver la acción desde la posición que desee, facilitando con ello la jugabilidad. Por medio de las teclas de función se le ofrece a al jugador una serie de vistas y por medio del ratón la opción de mover la cámara libremente.

Otra cuestión importante que corresponde tomar en cuenta es aumentar el rendimiento del juego. Se podrían utilizar un par de sistemas: controlando el nivel de detalle o LOD (Low of Detail) o creando otro juego de texturas de menor tamaño para objetos lejanos donde el detalle de su apariencia no es tan necesario.

3.2.5 ESTILO GRÁFICO.

Es evidente planear unos gráficos de acuerdo con la historia del juego. Si, por ejemplo, nuestro juego trata de batallas de guerreros medievales, los gráficos tendrán que ser propios de la época. Habrá que diseñar espadas y hachas en vez de pistolas láser.

Zone of fighters se desarrolla en un mundo espacial imaginario sin ningún antecedente histórico, pero si se sabe que existe una tecnología avanzada. Habrá grandes superficies al aire libre con vegetación, ríos, volcanes, nubes y sol. Habitarán seres de varias clases. Se dispone de naves de combate y distintos tipos de armas. Además, habrá esparcidas edificaciones como almacenes, generadores de energía y puentes. Toda está apariencia gráfica puede ser diseño libre.

Ciertamente, a todo esto tenemos que añadirle los efectos especiales, los más reales y espectaculares posibles. Se incluirán efectos climatológicos como lluvia; también humo y fuego para los volcanes. Habrá disparos, explosiones y por último un poco de sangre para las criaturas letales que nos acompañarán en la acción. Para esté cometido, utilizaremos un completo sistema de partículas.

3.2.6 AMBIENTACIÓN SONORA.

Un aspecto importante es sin duda el sonido del juego. Con él, vamos a reforzar la temática que hemos diseñado, creando el ambiente adecuado.

Para empezar, a cada acción que hagamos en el menú –con el ratón- le aplicaremos un sonido; así confirmaremos al jugador su elección. Puede valer con el sonido de un clic o un pequeño pitido. También sonará de fondo un tema musical con cierto ritmo pero de índole militar, invitando a la competición.

Una vez dentro del juego, los efectos de sonido se tornan más importantes, porque pueden ayudar en gran medida a resolver las situaciones. Es importante recordar que se debe aprovechar las características del sonido 3D para mejorar el propósito del juego.

Dotaremos a nuestra “bio-nave” de sonido con tono ascendente en la frenada. Para el disparo haremos distinción dependiendo el tipo de arma que se utilice y para el escudo de energía, un sonido envolvente y algo denso, además de de un pequeño aviso sonoro cuando su función este llegando a su fin. De esté modo, el jugador dispondrá de tiempo para reaccionar, por ejemplo, en medio de un combate.

Se podrá aplicar otra clase de avisos sonoros en situaciones como: nivel bajo de energía, la obtención de bonos o cualquier tipo de ítems (munición, arma, energía, escudo, etc.) asimismo, en cualquier contacto detectado por parte de las bio-navas con otro objeto del escenario (edificaciones, árboles, montañas u otros jugadores) se producirá la respuesta sonora correspondiente. No olvidemos los efectos sonoros, generalmente explosiones, que los impactos por disparos causan en la superficie, demás elementos del terreno u objetos.

Un detalle cada vez más extendido que ayuda mucho durante el desarrollo del juego es conducir al jugador por medio de una voz de fondo o voz en off, bien recordando el manejo de la bio-nave, avisando de la cercanía de peligros o en cada acción del jugador como elección de armas o acierto en el disparo.

En esté tipo de juegos, el desarrollo puede volverse bastante activo, debido al combate y la lucha con los demás habitantes de la zona. Sería apropiado acompañar toda está acción con música movida, con bastante ritmo, para suministrar aún más emoción al jugador.

3.2.7 COMPLEMENTOS DEL JUEGO.

Se debe tomar en cuenta la necesidad de acompañar al juego de complementos que ayuden a obtener un mayor interés por parte del usuario. Por

ejemplo los editores de niveles. Generalmente estás aplicaciones forman parte de las herramientas que los desarrolladores poseen para facilitar su trabajo.

Suelen ser de diseño un tanto complicado y de uso muy personal, y en pocas ocasiones están preparadas para servir al público. El gran interés del jugador hacia este tipo de aplicaciones ha propiciado que las compañías hagan un desarrollo paralelo de sus herramientas para incluirlas en sus lanzamientos. Además de los editores de niveles, como Q3radiant para la saga Quake o Worldcaft para HalfLife, debemos sumar visualizaciones de modelos, como el que ofrece el juego Neverwinter Nights o sistemas para crear tus propios MODs (un MOD permite tener acceso al código, para realizar cambios en él. Existe la modalidad de un jugador o multijugador, esto para crear nuevas modalidades de juego).

En definitiva, en está entrega hemos convertido un montón de ideas sueltas en un diseño tangible, real y práctico. Ahora él proyecto de video juego va adquiriendo forma, y la imagen que se tenía del mismo en la mente se debe plasmar en papel por escrito.

La tarea es diseñar cada elemento que compondrá él juego, con más detalle. Se deberá realizar dibujos o bocetos de todos los aspectos del juego para facilitar el diseño gráfico. Se describirá el comportamiento de cada componente y mecanismo de sus acciones. Se fabricara herramientas propias de desarrollo como visualizadores de terrenos y objetos o editores de zonas de combate. Y por último se llevara todo a la práctica, a la vida real, desarrollando el código.

3.3 ZONA DE GRÁFICOS.

La calidad gráfica del video juego ha cambiado siempre de forma paralela al desarrollo tecnológico de la informática. No hace más de diez años, era imposible

ver más allá de 16 colores en una PC y, prácticamente, el 3D se limitaba a líneas y modelos sin textura alguna.

Con la llegada del sistema operativo Windows de Microsoft, se abrió una nueva puerta para la evolución de la infografía que afectó inevitable mente al desarrollo de los video juegos para PC.

Se consolidó aún más la PC como ordenador para uso multimedia y empezaron a aparecer tarjetas gráficas mejores al servicio de microprocesadores cada vez más potentes.

En un principio, las poderosas herramientas gráficas estaban a disposición de grandes producciones cinematográficas, la televisión o la publicidad; sin embargo, a medida que evolucionaban las capacidades multimedia de los ordenadores personales, los juegos también lo hacían y su desarrollo demandaba más y mejores paquetes infográficos.

Las primeras herramientas gráficas de uso doméstico se utilizaban para dibujar y manipular gráficas de uso doméstico se utilizaban para dibujar y manipular gráficos en dos dimensiones. Generalmente, constaban de una ventana o parte de la pantalla donde era posible realizar operaciones de dibujo, y de un grupo de iconos que representaban las herramientas disponibles para el usuario: pinceles, goma para borrar, selección de colores y posibilidades de dibujar polígonos tales como rectángulos, círculos, etc.

Estos programas eran utilizados por los creadores de juegos para sus gráficos y existieron en sistemas domésticos anteriores a la maduración multimedia de la PC. El más conocido fue Deluxe Paint para los ordenadores Commodore y se convirtió en el precedente de los actuales programas de dibujo.

Con la llegada del 3D a los video juegos, la demanda de herramientas gráficas aumentó considerablemente. Además del tradicional y necesario programa de dibujo y retoque fotográfico, era primordial utilizar aplicaciones para la creación de objetos poligonales.

Aparece la necesidad de dar apariencia a estos modelos 3D por medio de texturas. Estos elementos gráficos pueden ser dibujados fácilmente con cualquier programa gráfico 2D; aun así, emergen utilidades que permiten dibujar directamente sobre el objeto 3D, facilitando la labor del grafista.

El mercado empieza a inundarse de herramientas 3D al servicio del desarrollo de infografía lúdica, apareciendo una gran oferta en calidad y en precio para el profesional y usuario de a pie.

3.3.1 PROGRAMAS QUE SE UTILIZAN.

Hoy en día, las producciones de video juegos han alcanzado una altura grandísima de calidad gráfica y precisan herramientas especializadas. A pesar del gran número existente de ellas, la mayoría cumple con los requisitos necesarios, y con su precio e implementación en el mercado lo que las diferencian. Debemos distinguir aquellas aplicaciones diseñadas para un uso en el campo 2D y 3D.

3.3.1.1 GRÁFICOS 2D, DIBUJO Y RETOQUE FOTOGRÁFICO.

Hay numerosas aplicaciones potentes para el diseño gráfico 2D, pero muy pocas han gozado de la fama necesaria para llamar la atención profesional y amateur. Sólo las grandes casas de desarrollo de software pueden competir en el mercado e implantar estándares. Tenemos, pues, el programa Photoshop de Adobe, PhotoPaint de Corel o Paint Shop Pro de Jasc Software.

Adobe Photoshop es el líder mundial indiscutiblemente y el preferido por los grafistas de video juegos. Fue prácticamente, el primer programa de edición y retoque fotográfico profesional del mundo y su diseño ha creado escuela gracias a su fácil interfaz de usuario, la gran potencia de sus herramientas, un sistema de composición por capas y su revolucionaria técnica para deshacer y rehacer acciones, entre muchas otras.

Aun así, existe la competencia, y todas las demás aplicaciones poseen un diseño y potencia muy similares y son capaces de manipular todos los formatos de imágenes más comunes, luego pueden servir perfectamente para el propósito deseado.

Paint Shop Pro 7 a pesar de su costo bajo de 100 euros puede competir perfectamente en prestaciones con programas de lujo como Adobe Photoshop. Y se convierte en la herramienta 2D preferida del usuario.

3.3.1.2 GRÁFICOS 3D, MODELANDO Y ANIMACIÓN DE OBJETOS.

La lista de los programas de diseños de tres dimensiones que existen es para rellenar líneas y líneas enumerándolos, aunque actualmente en uso en el mundo de los video juegos sólo hay unos cuantos. Al igual que en el diseño 2D con Photoshop, ocurre el diseño 3D con el 3D Studio Max. Esté programa es ineludible el líder mundial en el sector lúdico. Su arquitectura abierta ha posibilitado que sus características estén en continuo aumento y sus aplicaciones se extiendan para cubrir todas las necesidades que el diseñador gráfico de un juego precisa. Su formato de archivo genérico 3DS es un estándar mundial y rara es la aplicación o herramienta de programación 3D que no lo soporte. Es un programa ideal para modelar y animar objetos para juegos y también, por ejemplo, para generar películas animadas de presentación.

Hay otras aplicaciones 3D que se han adaptado perfectamente y con buena nota al diseño gráfico de juegos. Se pueden hallar herramientas de gran poder utilizadas en el mundo del cine o la televisión, como LightWave, Maya, SoftImage o Cinema 4D XL, y otras también profesionales pero más accesibles como Caligari TrueSpace o Bryce.

En ámbito profesional, las indiscutibles prestaciones de modelado y animación de LightWave y su relación calidad-precio, han contribuido a que pueda acompañar al 3D Studio Max en el ranking de herramientas más usadas en el desarrollo de video juegos.

Cinema 4D XL7 también compite con las aplicaciones más utilizadas del momento, como 3D Studio Max o LightWave, gracias a su robustez y versatilidad. Asimismo, la versión XL7 del Cinema incorpora el BodyPaint 3D, un sistema para el pintado de objetos muy potente.

Caligari, con sus versiones 4 y 5 del TrueSpace, ha encontrado un hueco importante en el diseño 3D para juegos, gracias a un acercamiento al usuario menos profesional –necesitando de una herramienta fácil de aprender, de bajo costo y con características competitivas-. Además, es de las pocas herramientas de diseño 3D que incorpora –de serie- la importación y la exportación de modelos en formato DirectX (extensión de archivo X), muy utilizado en el desarrollo de juegos para Windows.

Ha cautivado a diseñadores de todo el mundo gracias a su versatilidad, su potencia y su intuitiva interfaz de usuario. Integra todas las funciones necesarias para el modelado y animación de objetos 3D, ideales en el desarrollo gráfico para juegos.

El programa Bryce, creado por la desaparecida Metacreations, se ha convertido en un estándar en la creación de mundos artificiales y, hoy día, es

extraño encontrar un grafista de video juegos que no conozca sus fantásticas posibilidades para la producción de terrenos. Su singular interfaz de usuario lo convierte en una herramienta muy instintiva y fácil de aprender. Carece de posibilidad de exportar objetos –a excepción de los terrenos 3D, que puede hacerlo en multitud de formatos-. También ofrece la posibilidad de guardar en disco una imagen del mapa de altura para incluirlo posteriormente en nuestros juegos.

Muchos de estos programas pueden ser adquiridos sin demasiado esfuerzo económico por grandes desarrolladoras o incluso por las de nivel medio, pero resulta más complicado para el usuario doméstico o para un grupo pequeño que, a duras penas, encuentra en el mercado para sus juegos. Una solución a esté problema está en aplicación 3D menos poderosas, pero tremendamente útiles y económicamente aceptable como el 3D Canvas, MilkShape 3D o Zmodeler. Se puede adquirir fácilmente a través de Internet.

3.3.1.3 TEXTURIZADO DE OBJETOS 3D.

Al margen de las aplicaciones 2D y 3D, se encuentran herramientas que mezclan los dos conceptos gráficos con el propósito de hacer más factible, si cabe, el texturizado o apariencia de los modelos tridimensionales. Permiten importar objetos 3D para luego poderlos pintar con herramientas de dibujo de una forma cómoda e instintiva. Entre los más utilizados están Deep Paint 3D y Painter 3D.

3.4 ZONA DE AUDIO.

Los de formato que se utiliza en el transcurso de la fabricación de un juego depende de la capacidad de técnica que se posea. Por tanto, es importante saber

qué calidad de sonido se quiere emplear. Ya que cada formato posee cualidades distintas de reproducción y consumo de memoria diferente.

Entre los formatos mas utilizados hoy en día en el mundo de los video juegos son:

3.4.1 FORMATO DE AUDIO WAV.

Este formato fue creado por Microsoft para su sistema operativo Windows. Suele ser el más utilizado en el mundo multimedia por su flexibilidad. Al no llevar ningún tipo de corrección es ideal para transformarlo a distintos tamaños y frecuencias por medio de editores y codificadores de audio. Lo más destacado de esté formato es la elevada calidad de audio que puede archivar, hasta una frecuencia de 44,1 Khz. a16 bits en estéreo. Sin embargo, esté rendimiento tiene un precio: el excesivo tamaño de los archivos que se generan y la cantidad de memoria que consumen.

La pregunta es: ¿en qué puede afectar al rendimiento de un juego el que la música, en formato Wav, tenga un fichero de gran tamaño? Pues bien, al cargar el archivo desde nuestro programa, lo hacemos directamente en la memoria. Un archivo Wav con calidad máxima (44 Khz., 16 bits estéreo) prácticamente ocupa 10 Mb por minuto de audio. Así que, si se muestra música tiene una duración de 3 minutos, al cargarse en memoria ocuparía la terrible cantidad de 30 Mb, y eso sólo para la música. Para reducir el tamaño de los archivos Wav (independientemente de la duración) se suele disminuir su calidad de forma aceptable, reduciendo su frecuencia.

Generalmente, por esté motivo, esté formato se suele usar para sonidos que irán en el juego y no en la música. Además, estás muestras se suelen reducir a calidades distintas dependiendo de su uso. En efectos especiales como

disparos, explosiones, etc., es permisible una reducción de frecuencia a 22 Khz., y para las voces incluso se puede prescindir del estéreo y convertirlo a mono, ahorrando así la mitad del tamaño que ocupa. Bajar hasta 11 Khz. puede producir gran cantidad de ruido e incluso cortes en la reproducción, por lo que no es aconsejable.

No obstante, este formato de audio ofrece una gran ventaja: no necesita ser descomprimido, por consiguiente, no resta rendimiento al juego, un hecho que se debe tomar muy en cuenta a la hora de aplicar sonido al video juego.

Existe una estupenda opción al problema del tamaño en formato Wav; y es que se puede utilizar otro formato que comprima los datos sin perder prácticamente calidad. Esto hace referencia al Mp3.

3.4.2 FORMATO DE AUDIO MP3.

El Mp3 despegó en 1999 cuando Shawn Faning muchacho de 18 años, frustrado por pasar hasta 60 horas en la computadora buscando sitios para bajar música, ideó una aplicación conocida como Napster, que permitía compartir archivos para que cualquier persona conectada pudiera tener acceso a una especie de base de datos en línea tan grande como la cantidad de usuarios y de archivos que estuvieran en línea en el momento.

Esté paraíso no duraría mucho y fue en 2000 cuando Shawn se dio cuenta de que su idea cambiaría para siempre la forma de distribuir la música. La RIAA (Record Industry Asociation of America) lo demandó junto a la compañía distribuidora y desarrolladora y desarrolladora de Napster, alegando que la distribución de Mp3 mediante esté servicio gratuito era ilegal y no pagaba regalías, derechos de autor o cuotas a los artistas y compañías discográficas.

El Mp3 como tal no representa ninguna violación a la ley, de hecho, éste es solamente el formato de audio en el cual los usuarios comúnmente pueden grabar CDs a sus computadoras para después escucharlos ahí, o bien en aparatos compatibles con el formato.

Esté formato es el rey actualmente, debido a la reproducción del tamaño de los archivos sin pérdida de calidad. Funciona por medio de algoritmos o funciones que eliminen las frecuencias que el oído humano no puede percibir. Es capaz de mantener una calidad de disco compacto (44 Khz., 16 bits estéreo) reduciendo el archivo hasta 14 veces.

No podía faltar para toda está maravilla un inconveniente, un sistema de decomprensión para poder ser reproducido. Utilizar ficheros Mp3 para la música o los efectos de sonido en un video juego provocaría una bajada del rendimiento, debido a que tienen que ser descomprimidos en tiempo real a medida que se reproducen. Asimismo, aumentaría los requisitos del ordenador.

Existen otras opciones para dotar a un juego de música con poco consumo de memoria y velocidad; son los ficheros de música MIDI, MOD, S3M o XM para secuenciadores.

3.4.3 FORMATO MIDI PARA MÚSICA.

En esté tipo de archivos se puede almacenar secuencias de música en forma de eventos MIDI. Ocupan muy poco espacio porque no poseen información de audio sino de caracteres, más concretamente valores numéricos o parámetros propios del protocolo MIDI, como son el tipo de nota, la duración, etc. Estos valores son interpretados y provocan sonido a través de la tarjeta de audio del ordenador. Es fácil, pues, transportar la música de un sistema a otro. Además, estos eventos pueden modificarse desde el programa en tiempo real; y lo mejor de

todo, no requieren un gran sistema para ser reproducidos. El inconveniente radica en que estos archivos no guardan el timbre, por lo que los sonidos empleados para la música están limitados a los que posee la tarjeta de sonido, y no todas las computadoras contienen la misma.

También hay que decir que aparte de no tener una calidad estándar, los ficheros MIDI no incluyen la reproducción de muestras, como loops de batería o voces, reduciendo la intención creativa. Para poder manejar secuencias formadas por muestras se pueden utilizar formatos MOD, XM o S3M.

3.4.4 FORMATOS MOD, S3, M Y XM DE MÚSICA PARA TRACKERS.

Estos formatos son muy utilizados para proporcionar música a los video juegos, debido a su relación tamaño-calidad-rendimiento necesario. Cada uno de ellos surgió de un secuenciador distinto y difieren unos de otros en prestaciones como número y calidad de las muestras, limite de filas por patrones, número de efectos permitidos, etc.

El formato MOD es el que posee más limitaciones, por ser el más antiguo; aun así, está muy extendido y es el más soportado por todos los secuenciadores.

3.5 BLITZ 3D.

Investigación Ltd del Blitz es una compañía de desarrollo del software dedicada a traer herramientas y utilidades de la creación del juego.

El Software del buque insignia es Blitz3D, el lenguaje de programación de los juegos del golpe 3D usado por millares de programadores alrededor del mundo.

Utilizado crear juegos, sin ser frustrado por la complejidad de lenguajes de programación tales como C++ o Java.

Blitz3D proporciona un ambiente simple con un gran alcance para la creación del juego - simple, porque está basado alrededor del popular y fácil de utilizar lenguaje de programación BASIC; y de gran alcance, gracias a un motor subyacente altamente optimizado 2D/3D.

Blitz3D incluye muchos comandos de ayuda para la creación del juego, el sistema de comando de Blitz3D's se ha diseñado cuidadosamente para proporcionar la máxima flexibilidad con el mínimo esfuerzo.

Se ha logrado escribir juegos de la calidad comercial en Blitz3D, grandes títulos como el país de las maravillas y Juno Némesis fueron escritos en Blitz. De hecho, muchos autores del Shareware se están trasladando a Blitz desde idiomas tales como C++ debido al tiempo de desarrollo más rápido.

Blitz3D se diseña alrededor de la idea de entidades. Típicamente, un juego 3D consistirá en muchos tipos de los elementos 3D tales como cámaras fotográficas, luces, acoplamientos etcétera. En Blitz3D, éstas todas se consideran ser entidades. Esto significa que los mismos comandos se pueden utilizar para manipular y para ocuparse de una amplia gama de los objetos 3D - por ejemplo, el comando de TurnEntity se puede utilizar para dar vuelta a una cámara fotográfica, a una luz, a un acoplamiento, o aún a un terreno masivo.

Los acoplamientos son el workhorse de cualquier motor 3D. El Blitz permite crear acoplamientos propios del rasguño, o los acoplamientos existentes de la carga en formato de X, de 3DS o de B3D. El formato de B3D es único a Blitz3D, y se ha diseñado para prever todas las características del motor de Blitz3D. Los exportadores de B3D están disponibles para una amplia gama de modelar software incluyendo el estudio 3D, la onda luminosa y Truespace. Los

acoplamientos de Blitz3D proporcionan características tales como colores de la cima y la coordenada múltiple de la textura fija.

Los terrenos de Blitz3D se pueden utilizar para generar escenas al parecer ENORMES. Esto se alcanza con un truco conocido como reducción de LOD (nivel-de-detalle). Este proceso totalmente automático trabaja escogiendo un sistema de polígonos que aproxime la forma de un heightmap muy de alta resolución. Los terrenos de Blitz3D se pueden también modificar en el tiempo real - grande para los agujeros que soplan en cosas.

Sprites es simplemente 2.os rectángulos textured - fáciles de instalar (setup) y grande para las cosas como efectos de la partícula, cabeza-para arriba-exhibe etcétera. Sprites se puede instalar para orientarse automáticamente para hacer frente a la cámara fotográfica, dando la ilusión de un objeto esférico, pero sin los gastos indirectos de dibujar una tonelada de polígonos.

Las cámaras fotográficas de Blitz3D ofrecen a programador control directo sobre el viewport, zumbido, gama que acorta, efectos de la niebla e incluso ofrecen un modo isométrico. Las cámaras fotográficas múltiples no son tampoco problema – se puede utilizar simplemente CreateCamera tantas veces como se requiera.

Blitz3D ofrece luces direccionales, luces del punto y luces del punto. Se puede controlar el color, la gama y “el cuerpo” de luces con facilidad.

Blitz3D hace el uso extenso de la idea de cepillos de colorear, texture y manipula de otra manera el aspecto de entidades. Los cepillos ofrecen una amplia gama de efectos tales como toques de luz del teñido, el multitexturing y specular.

Multitexturing permite que usted aplique hasta 8 capas de textura a una entidad. Esto se puede utilizar para alcanzar los efectos tales como lightmapping,

el detalle texturing, toques de luz specular y un anfitrión más. El sistema multitexturing en Blitz3D también permite controlar un exacto de texturas que se combinan juntas.

Detrás del ambiente se crea la ilusión de reflexiones adentrándose a una escena. Blitz3D proporciona 2 formas detrás del ambiente - esférico o cúbico. Los mapas cúbicos del ambiente se pueden incluso poner al día en el tiempo real, previendo efectos impresionantes, dinámicos de la reflexión.

Todo lenguaje de programación que trabaje bajo el entorno Windows como es el caso de Blitz 3D precisa de algo más que para escribir código. Necesita un completo entorno de programación que además compile, ejecute y depure los programas.

Ciertamente, el corazón de Blitz 3D es una herramienta de estás características que ofrece una interfaz amigable. Además incluye un sistema de ayuda interactivo completo en estilo HTML con multitud de descripciones y ejemplos.

3.5.1 USO DEL EDITOR.

Una vez ejecutando Blitz 3D (B3D) aparece el menú principal del archivo de ayuda (help). Esté menú está en formato HTML y se navega por él como si en una página Web. A través de los hipervínculos (links), en color, se accede a todas las secciones. Además, para facilitar la navegación, el editor incorpora tres sencillos botones en la barra de iconos situada debajo de los menús desplegables. El botón inicio sirve para ir al index (índice) y flechas ayudan a navegar hacia delante o atrás, para ir a las páginas de ayuda ya visitadas.

Si se pulsa en “samples” podemos cargar desde disco directamente al editor del de programas, cualquiera de los ejemplos que nos proporcionan. En “Command Reference” se encuentra la descripción y uso de todos los comandos de B3D y en “Language Reference” una descripción del funcionamiento del lenguaje.

Antes de continuar, se puede disfrutar por un momento de los ejemplos de B3D. En primer lugar se pulsa en “samples”, aparece una ventana de Windows con una lista de todas las carpetas de ejemplos. Se puede elegir un ejemplo que está dentro de está, al pulsar abrir aparecerá el editor de programas con el código del ejemplo.

Se puede ejecutar el ejemplo de tres formas distintas. La primera es pulsando el botón con la acción run, si se deja el puntero un momento sobre el icono podrá mostrar la función. La segunda forma es yendo al menú principal, desplegando “Program” y eligiendo la opción “Run Program”. Y la tercera es pulsando “F5”; evidentemente, es la más fácil.

El entorno B3D, puede tener en el editor más de un programa a la vez. Cuando se realiza está función se puede intercambiar la visión del programa en la pestaña con el nombre del programa o mediante CTRL+SHIFT+TAB una vez, CTRL+TAB dos veces.

Es posible cargar cuantos programas se quiera y disponer de todos ellos en el editor. Esto ayudará cuando se decide dividir el programa principal en subprogramas más pequeños para facilitar la corrección de errores o una posible ampliación. Un ejemplo práctico de está procedimiento, sería tener un fichero, con un programa que manipulara sólo el control de la nave por el jugador. Y para que todo funcione, el primer programa (el principal), incluiría una llamada al segundo.

3.6 3D GAME MAKER.

Funcionamiento: 3D Game Maker es una aplicación que permite diseñar un juego completo utilizando el famoso sistema Filmation. Todo el proceso se realiza de forma sencilla a partir de 3 programas que son los que componen el paquete: Graphic Editor 3D (para el diseño de los gráficos del juego), Room designer 3D (para diseñar todo el mapeado del juego, colocando objetos y enemigos habitación por habitación) y Game Maker 3D para compilar el programa final y guardarlo. Este último, incorpora también un sencillo juego creado con el paquete, para enseñar lo que se puede hacer con él.

- **Título:** 3D Game Maker.
- **Año:** 1987.
- **Compañía:** CRL Group PLC.
- **País:** Reino Unido.
- **Autor/es:** Iain Christopher Hayward.
- **Distribuidora en España:** Dro Soft, Serdisco.
- **Tipo de programa:** Diseño de juegos en 3D.
- **Multicarga:** Sí (3).
- **Versiones:** 48k.

Es un deleite, un sueño hecho realidad para todos aquellos que alguna vez soñaron con crear su propio juego al más puro estilo Knight Lore. Es fácil de usar y muy intuitivo.

Apuntes: El creador de esta joya, nunca destacó como programador de juegos, algo curioso teniendo en cuenta el gran nivel del programa que nos ocupa.

Game Maker es una potente herramienta para crear tus propios juegos de forma sencilla, visual y sin conocimientos de programación.

Podrás diseñar los juegos con tus propios gráficos, sonidos y efectos o utilizar los que incorpora Game Maker en su editor. El programa es capaz de abordar casi cualquier género, plataformas, puzzles, acción e incluso juegos 3D tipo Doom.

Como resultado de esté capítulo se llega a la solución de que el video juego desde el inicio de su historia hasta nuestros días es motivador de chicos y grandes para jugar en sus diferentes juegos y plataformas que existen en el mercado, se reviso la gran competencia entre Nintendo, Sony y Microsoft en la creación de consolas para la gran diversidad de software de entretenimiento casero. Se reviso la complejidad de la elaboración de un video juego pues esté necesita de una gran creatividad, además de la utilización y el conocimiento de software para ejecución de gráficos, la implementación de los diversos formatos de audio que existen para el video juego, y de una manera un tanto más importante la forma de controlar todo esto mediante un lenguaje de programación para estructuración.

CAPÍTULO IV
REALIZACIÓN DEL VIDEO JUEGO



OBJETIVO.

Se aplicará el conocimiento en el desarrollo del video juego para lograr la creación de una estructura, la cual se presentara como el proyecto de nuestro software.

Esté es el último capítulo de una serie de cuatro, de los cuales, en esté se tendrá que exponer el caso practico en la elaboración de una propuesta de video juego, para lo cual se utilizo el programa Game Maker por su facilidad de uso y ya que no tiene una gran complejidad en su aplicación como se pudo explicar en el capítulo anterior, por lo cual se realizara una breve definición de cómo se puede materializar un juego en unos sencillos pasos con lo que se podrá obtener el proyecto de un video juego.

4.1 REALIZACIÓN Y PRESENTACIÓN DE UN VIDEO JUEGO.

En está tutoría se va ha ver los conceptos y puntos básicos que hay que se deben tener en cuenta a la hora de realizar un juego con 3D Game Maker. No es una tutoría paso a paso de como utilizar los programas, para eso ya existen manuales y tutoriales de 3D Game Maker que se localizan en la página "Web" del programa. Esta tutoría es un razonamiento filosófico y sobre la utilización del sistema de trabajo.

Muchos de los puntos de esta tutoría son comunes a la creación de juegos en general, ya sean comerciales, realizados con un "Game Maker" (programa para hacer juegos) o programados con un lenguaje de programación. Pero para el caso a centrar en el proceso concreto de elaboración de un juego con el entorno de desarrollo en 3D Game Maker, con sus ventajas y limitaciones.

Al final del trabajo se encuentran los pasos que a realizar para el desarrollo, como ejemplo: el juego de **"Pacman Proy"** en Game Maker, juego que no es ninguna maravilla de técnica pero que es un claro prototipo de muestra para un posible juego que se ha de desarrollar, ya que siempre es mejor tener un ejemplo y no solo que todo sean aburridos manuales teóricos sin nada práctico que demuestre que al menos funciona mínimo.

4.1.1 ENTORNO 3D GAME MAKER.

Normalmente antes de atacar la labor de realizar un juego se debe decidir el entorno del desarrollo que se a utilizar en función de la temática de juego que se desea realizar, de los medios de que se dispone (materiales y humanos), del tiempo, del presupuesto, etc., pero en este punto ya la decisión fue por 3D Game Maker, ya sea por lo que se puede ver y lograr con esté entorno, ó porque es un Freeware, ó ya que no hay que programar en Visual C/C++.

En la actualidad existen numerosos entornos de creación de juegos 3D disponibles que hacen más llevadero el proceso de creación de un juego de este tipo. Los hay con entornos de programación en Visual C, otros hay que programar pero su lenguaje es bastante sencillo y potente, otros no hay que programar pero hay que utilizar unos sencillos comandos para establecer la lógica del juego, otros son comerciales, otros Shareware, otros Freeware, otros con mucha comunidad en Internet, otros son editores de niveles de juegos comerciales (el creador del juego y el jugador han de tener el juego comercial para poder jugar al nuevo juego desarrollado), etc.

Como se puede observar hay sistemas para todos los gustos, bolsillos y niveles, 3D Game Maker se engloba en el tipo de sistema que no hay que programar aunque si es necesario aprender sus sencillos comandos, la versión 6.0 es completamente gratuita, no necesita de ninguna licencia comercial para funcionar.

Lo que en realidad debe quedar claro es que el juego que se ha de realizar no va a ser comercial ni va a rivalizar con Halo 2 ni con Doom III, por lo que se corresponde ahorrarse los estudios de mercado y la viabilidad del proyecto y demás temas comerciales.

4.1.2 TRABAJAR POR EL PROYECTO.

El juego a realizar es porque se quiere pasarla bien en el proceso de su elaboración, porque se ambiciona enseñar a los amigos, porque aspira presentarlo en un concurso (Trabajo Recepcional), porque persigue adquirir experiencia para realizar otros proyectos más ambiciosos o porque no se tiene nada mejor que hacer con el tiempo libre.

Existen otros artículos o tutoriales que también describen todo el proceso de creación de un juego, los cuales orientan hacia el tema comercial, pero se ha de ser más realistas visto el panorama actual, que por lo menos existe en México.

La verdad es que la industria de creación de video juegos para PC y consolas, es en la actualidad es muy limitada o casi imposible de ver en México, y no es por falta de gente con talento, pero esa es la cruda realidad. Por eso esté tutorial no pretende ofrecer falsas ilusiones haciendo creer que con una idea, 3D Game Maker y un tutorial se va poder dar el “golpe” en el mundo de los videojuegos y colocarse en un exitoso grupo de desarrollo ganando un montón de dinero y prestigio, después de haber vendido 2 millones de copias de un genial juego.

Por eso se ha realizar algo más posible para una aspiración, para ver lo que es enfrentarse a un proyecto (aunque esté sea modesto) y sobre todo porque tiene ganas de volcar toda la creatividad y talento en algo que gusta y agrada.

4.1.3 EXIGENCIAS PARA TRABAJAR.

Básicamente se puede reducir en 5 puntos:

- **Tiempo:** Pues algo de tiempo se tiene que dedicar al proyecto, y si esté puede ser a diario, mucho mejor, ya que si se va espaciando el tiempo que se le dedica, al final se puede caer en el desánimo y abandonar el proyecto. A parte de que si es un tiempo continuo se posee todo más fresco y a la larga se trabaja mejor y con más creatividad.

- **Ganas:** Para abordar proyectos por amor al arte, que no te da un sueldo ni te aprieta un jefe, es fundamental tener muchas ganas de realizarlo, ya que si hay pocas ganas lo más seguro es que el proyecto se termine abandonando al primer contratiempo o al aparecer en la vida otras cosas más importantes y prioritarias que hacer juegos.

- **Medios:** Evidentemente se requiera al menos un ordenador y diversos programas para poder trabajar en el proyecto. De los cuales con 3D Game Maker no es necesario un superordenador para hacer algo medianamente decente, con un Pentium III a 500 Mhz hay de sobra.

- **Conocimiento:** De nada sirve todo lo anterior si no sabe ni como realizaría el comienzo de un proyecto, ni como manejar medianamente bien los programas que se utilizarán.

- **Personal:** En la práctica se puede realizar el juego uno solo, pero siempre será más divertido si se realiza en equipo, y más si el equipo se logra formar de personas especializadas en distintas disciplinas.

En resumen:

- El tiempo, las ganas, el ordenador lo pone el diseñador del juego.

Los programas son: 3D Game Maker, y un editor 2D tipo Gimp o Adobe Photoshop para retocar texturas.

Los conocimientos también son los que ya se poseen.

El personal necesario puede ser:

- Director de proyecto.
- Programadores.
- Guionistas.
- Grafistas 2D.
- Grafistas 3D.
- Modeladores 3D.
- Animadores 3D.
- Creativos.
- Músicos.

Eso es lo que básicamente se necesita para crear un juego comercial, y eso sin contar con los editores y distribuidores.

Como 3D Game Maker es un entorno de desarrollo que no necesita programadores, pues eso se ahorra, pero de todas formas hay que aprender unos comandos básicos que se tienen que dominar si se quiere implementar algún tipo de trama al juego.

4.1.4 NO DISPONIBLE CON GAME MAKER.

Trabajar con un “Game Maker” o programa para realizar juegos, tiene sus ventajas y desventajas, como todo en está vida. A parte de evitarte programar,

existen muchas tareas que ya están automatizadas en parte o completamente, pero por otro lado tienen una serie de limitaciones que tienen que asumir.

A las que de referencia existen que si se estuviera por ejemplo realizando un juego con Visual C/C++ y si en el juego tiene que aparecer la siguiente escena: “un coche se aproxima derrapando al jugador, se queda a 2 centímetros de él y del interior del coche sale una chica deslumbrante que te dice:” ¿Entras?, a lo que el jugador podrá ponerse al volante y tomar el control del coche, pues al programar se decide que hay que hacer eso, programador implementa en poco tiempo una flamante rutina, optimizada, recursiva, prototipada y orientada a objetos en lenguaje “C++” que satisface las exigencias de un guión.

Pues bien, el caso anterior es muy bonito pero imposible de realizar con 3D Game Maker, de aquí viene la necesidad de conocer muy bien las cualidades y limitaciones de 3D Game Maker antes incluso de realizar el guión de un juego.

4.1.5 TEMÁTICA DE UN JUEGO.

3D Game Maker te limita a ciertos tipos de juegos. Básicamente permite realizar:

- Juegos 3D de disparos en primera persona, tipo Quake o Doom.
- Juegos 3D de disparos en primera persona con trama, tipo Half-Life.
- Juegos 3D de disparos en tercera persona con trama, tipo Max Payne 2.
- Juegos de aventuras 3D, tipo Tomb Raider.

O juegos con características combinadas de diversos géneros existentes en el mercado o a realizar con la creatividad y la herramienta que es Game Maker para ir desenvolviendo un videojuego.

Con esto no se quiere decir que se pueda hacer un juego igual que los mencionados, pero si pueden orientar para saber el tipo de mecanismo de juego o la vista que el jugador tiene del escenario.

Así que de momento se puede ir olvidando la idea de hacer juegos RPG (Rol), RTS (Estrategia táctica en tiempo real), deportivos, de lucha, de coches, de aviones o de naves interestelares (complejos).

4.1.6 EL GUIÓN.

Una mala costumbre que hay a la hora de hacer juegos “amateur” es que se comienza con muchas ganas y directamente ya se sienta la persona delante del ordenador a 'hacer cosas” e ir enganchándolas sobre la marcha para que todo vaya encajando según la “visión mental” del creador del juego.

Lo cual es un gran error, porque efectivamente se pueden realizar juegos así, pero como resultado final es algo inexplicable y la gran diferencia que hay entre trabajar sin guión a trabajar con guión es muy extensa para la realización de un buen proyecto.

También es útil en el caso de que se cumpla una temporada para el desarrollo del juego, con un guión sólido y detallado siempre te será más fácil retomar el proyecto.

Como se puede y deberá prestar atención, es muy aconsejable que se elabore un guión del juego, por lo menos con trama de principio a fin, para tener siempre a mano una guía de lo que ha de hacer tu juego. No es necesario que se deba realizar un “Story Board” de cine, ya que es un proyecto sencillo para entretenimiento y un trabajo recepcional, pero se puede llevar acabo la secuencia de pasos siguientes:

- **Introducción:** Texto que podrá servir para la introducción del juego y que sitúe un poco al jugador.

- **Desarrollo del juego:** Textos que expliquen las situaciones del juego y los personajes que intervienen. Textos que expliquen que interacción hay entre los personajes y el jugador. Que dicen, que hacen, que piden, que dan, donde están, cuando aparecen, cuando desaparecen. Textos que expliquen que interacción hay entre el jugador y partes del escenario (paneles, palancas, botones, pedestales, pozos, etc.). Textos que expliquen que objetos utiliza el jugador, donde los encontrará, donde los utilizará, de quien o que los recibirá, porque los necesitará, que produce el objeto utilizado. Si el juego es lineal, o sea, que el jugador ha de ir haciendo una serie de tareas en un orden para poder seguir avanzando, se deberá escribir los textos anteriores en el mismo orden que se producirán.

- **Personajes secundarios:** Nombre de cada uno de los personajes que intervendrán en el juego sin ser el personaje protagonista ni los enemigos que disparan. Corresponde dejar un cierto espacio para poder añadir posteriormente que fichero 3D será cada personaje.

- **Personajes protagonistas:** Nombre del personaje o personajes que manejará el jugador. Deja espacio para poder añadir posteriormente que fichero 3D será cada personaje.

- **Personajes enemigos:** Aquí más que nombres de personajes, se refiere a personajes que dispararán al jugador y que se desplazarán por el escenario de juego. Con darles un número del 1 al 8 será suficiente para de momento referirse a ellos. Dejar espacio para poder añadir posteriormente que fichero 3D será cada enemigo.

- **Objetos:** Nombre de los objetos que el jugador podrá utilizar. Darles un número (a partir del número 2) y un nombre. Dejar espacio para poder añadir posteriormente que fichero 3D será cada objeto.

- **Otros modelos:** Aquí es necesario dejar espacio para poder poner los nombres de otros ficheros 3D que puedan necesitar dentro de tu escenario y que no sean propiamente objetos, ni personajes.

- **Esquema del mapa:** No estaría de más elaborar un simple esquema del mapa del escenario. Es suficiente dibujarlo a base de cuadrados que representan las distintas zonas donde evolucionará el jugador, que puertas conectan unas zonas con otras y que se necesita para acceder a las zonas.

Dentro de esas zonas puede indicar que objetos y personajes se contiene, de esta forma de un vistazo podrá ir estableciendo el recorrido que tendrá que hacer el jugador, con quien tendrá que encontrarse, como podrá acceder a ciertas zonas donde tendrá que recoger objetos que le permitirán acceder a otras zonas o utilizarlos con alguien o para hacer algo.

Además de un guión, esta estructura sirve de base para saber que elementos intervienen en todo momento en tu juego. Lo ideal es que se comience por la “Introducción” y el “Desarrollo” y según vayas avanzando con ello vayas rellenando el resto de apartados del guión.

Aún no ha conectado el ordenador, con esto se hace referencia ha que todo este proceso de planificación y estructuración se puede realizar perfectamente con papel y lápiz, tumbado en la playa, en un parque o en el autobús. Evidentemente tampoco va ha planificar al 100% de todo el juego, eso es normal, por lo que tampoco será cosa rara que durante el desarrollo del juego se tendrá que retocar el guión sobre la marcha (pasa hasta en el rodaje de las películas), pero el caso es

que obtenga siempre un guión del cual echar mano. Pero hacer eso, redefinir el guión y emplear el cambio en el juego, o al revés, pero lo que no se tiene que hacer es cambiar algo en el juego y no reflejarlo en el guión.

4.1.7 DESARROLLO DEL GUIÓN.

Hay dos formas de desarrollar un guión:

Primera forma: Tener en mente una maravillosa historia que se escribe muy bien en un guión y luego buscar los medios para llevarla a cabo. Si es necesario se deben crear los medios y también se crean todos los objetos, personajes, animaciones, texturas, mapas, sonidos y músicas que requiere tu idea.

Segunda forma: Evaluar lo que se tiene al alcance, con lo que se haya podido conseguir, Freeware por Internet y en base a eso se escribe el guión.

El primer caso es una tarea “faraónica” y según que situaciones se planteen en el guión será del todo imposible realizarlas con 3D Game Maker.

En el segundo caso, es más sencillo y está al alcance de casi todo el mundo, aunque tampoco hay que descartar crear alguna que otra textura o alguna que otra animación.

De acuerdo que al final será un refrito de cosas recogidas por Internet, pero si se logra elaborar una buena trama para el juego (ya sea misteriosa, divertida, surrealista, original, emocionante, etc.) y creas nuevas texturas para los personajes y escenarios, nadie podrá dejar de ver el mérito al crear algo nuevo aún reciclando elementos de distintas fuentes.

Pero se decidirá a que nivel de creación se implica en el desarrollo. Evidentemente si dispone de una persona especialista en modelación y animación, pues se podrá crear objetos y personajes completamente nuevos, o si dispone de un músico incluso podrás ponerle una exclusiva banda sonora a tu juego.

El consejo en el caso de que sea el primer juego con 3D Game Maker es que se cree de la segunda forma, eso dará experiencia sobre las herramientas, será más rápido y se tendrá menos posibilidades de estrellase en un proyecto excesivamente ambicioso y laborioso.

En el caso del guión del juego de ejemplo **"Pacman Proy"** en Game Maker, lo que se realiza fue lo siguiente por este orden:

- Analizar las posibilidades y limitaciones de 3D Game Maker.
- Buscar un escenario gratuito, le cambiar las texturas y decorarlo al gusto según la idea que se tenga en mente para el juego.
- Buscar personajes y objetos que haya disponibles en formato .jpg ya que con Game Maker se pueden animar las secuencias del juego cuadro por cuadro (frames).
- Hacer el guión según lo que tenga disponible, pero siempre con el concepto del juego presente.

4.1.8 ORDEN Y DESORDEN.

Si a la hora de realizar el juego no se sigue un orden en las tareas de creación es muy posible que se pierda mucho tiempo o que cosas que se habían realizado, ahora al haber llegado a un punto concreto del desarrollo, resulta que han de ser de otra forma para que las cosas tengan sentido.

Se puede seguir el siguiente orden para crear un juego como: **“Pacman Proy”:**

1. Se Carga un escenario (en formato .bmp), utilizando la opción “ADD/ADD Background, y posteriormente añadiéndolo como Room”.

2. En el escenario eliminan numerosas entidades que no existen en 3D Game Maker que venían importadas del escenario.

3. Aumenta en el fichero de Room los nombres de texturas que no tiene, ya que estos niveles vienen sin texturas, pero si que cada pared, suelo o techo tiene asociado su nombre de textura. Sólo hace falta que encuentre esos nombres de texturas en su fichero de Room al cargar el nivel y entonces ya veremos esas texturas aplicadas en el escenario en los lugares que les corresponden.

4. Compilar el escenario y caminar por él, como el que pasea por una casa que va a comprar, reformar o decorar, fijándome en todos los detalles para poderlos explotar a la hora de crear el guión

5. Se colocaran unos cuantos personajes en formato .bmp y convertirlos, al a realizar acción con la opción Edit Sprite para movimientos de la forma, los que pueden parecer que poden servir para el juego.

6. También se colocan algunos objetos en .bmp. Para objetos que servirán para que los pueda recoger el jugador al recorrer el escenario, como es el caso de bonus.

7. Ahora es cuando me la elaboración del guión pensando en lo que se pude hacer con 3D Game Maker y lo que no se consigue. Cabe hacer mención

que con imágenes .jpg, y .bmp se logran animar, y dar forma a personajes y zonas.

8. Se prueba al personaje protagonista para ver como se ve dentro del escenario y que rendimiento daba el motor 3D. Esto siempre se hace al probar nuevos objetos o personajes dentro del escenario, ya que la inclusión de alguno de estos modelos 3D podría afectar considerablemente al rendimiento general del juego debido al excesivo número de polígonos de que constaban ciertos ficheros.

9. Una de las tareas en **“Pacman Proy”** es recolectar las bolitas amarillas para conseguir punto, al conseguir las esferas de mayor tamaño los enemigos se podrán atacar y eliminar temporalmente con un porcentaje de store para el jugador.

10. Todo eso lo realiza en cuatro escenarios o niveles de juego, donde se logra poner todos los objetos y los personajes implicados. Esto permite chequear el mecanismo del juego de una manera muy rápida. Cuando estuviera bien chequeado ya distribuiría los objetos y personajes a sus posiciones reales.

11. Se añade al escenario algún objeto, pero de momento dejé que se pudieran abrir siempre para poder alcanzar todas las zonas de manera rápida para chequear acciones que tenían lugar en esas zonas y checar el rendimiento del escenario.

12. La posición a colocar todos los objetos y personajes en las perspectivas que requería el guión y se ubican las condiciones necesarias para realizarlo de acuerdo al guión.

13. Probar el juego de principio a fin, sólo para comprobar que podía llegar al final haciendo todo lo que tenía que disponer.

14. Añadir unos cuantos enemigos con la única finalidad de entorpecer la buena labor del jugador y elevar más la dificultad de matar que había que eliminar para que permitiese el acceso a una zona.

15. Al parecer en un momento al pasarse de enemigos se formo una gran labor, por lo que al darse uno cuenta de que así sería imposible que alguien pudiera llegar al final del juego, por lo que se eliminan algunos enemigos. El resultado fue que ahora era demasiado fácil, al acabar con todos los enemigos y llegar al final del juego. El caso fue ir equilibrando el nivel de dificultad del juego hasta encontrar un punto adecuado de jugabilidad y dificultad.

16. Buscar por Internet efectos de sonidos cortos en formato WAV y diversas melodías en formato MP3.

17. Cuando se está orgulloso con el resultado de la jugabilidad del juego, se le añaden algunos efectos de audio que se producirán en momentos puntuales en el transcurso del juego.

18. Luego aumentar un fondo musical que sonará durante el desarrollo del juego, otro para el final de partida y otro para cuando el jugador llegara el final del juego.

19. Jugar el juego con todas sus características completas unas cuantas docenas de veces, lo que da acceso a permitir observar pequeños detalles que se deben acabar de pulir. Algunos son faltas de ortografía, pero otros eran más graves como un personaje que se suponía que no te dejaba

pasar, pero intentándolo atravesar pegado a la pared, el jugador conseguía pasarlo. Lo resolví estrechando más las paredes a los lados del personaje.

20. Dejé que algunas personas probaran el juego y que lo probaran a fondo, haciendo todo tipo de estupideces y barbaridades para que pudieran petar el juego o que intentasen saltarse alguna de las tareas que eran obligatorias para seguir avanzando.

21. Cuando el juego estaba lo suficientemente probado se vuelve a compilar.

22. Después se regresa a probar el juego en distintas máquinas observando que efectivamente el rendimiento había aumentado bastante.

23. Finalmente se da por finalizado todo el desarrollo del juego.

Game Maker es una herramienta con la que se puede crear juegos de ordenador, sin necesidad de tener amplios conocimientos de programación. Su interfaz de gran facilidad de uso permite que se pueda dar vida a versiones propias de juegos tipo Pac-man, Arkanoid, Tetris, etc. Pero, además de las copias, también se logra crear casi cualquier juego que se te ocurra del tipo de plataformas, laberintos, multijugador, isométricos, etc. El programa incluye una serie de imágenes y sonidos integrados en el juego para que tener material con el que empezar.

Con la terminación de esté capítulo se logro obtener el proyecto del video juego **“Pacman Proy”,** así como también se explico la elaboración del mismo, con lo cual se espera que pueda servir, como ejemplo de la estructuración de un video juego en el programa Game Maker, de está forma se hablo, que existen diversas maneras de constituir un video juego más especifico y mejor hecho en lenguajes de programación orientada a objetos, aunque en está condición puede llegar a ser más complicado, pero el resultado es sin duda es de mayor expectativa.

CONCLUSIÓN.

Se ha revisado y analizado a través del primer capítulo cuidadosamente los inicios de la computación, las consecuencias y la gran magnitud con la que ha influido está poderosa máquina.

En esté trabajo se examino el funcionamiento de un ordenador mediante el hardware y, toda su estructura física, de la cual se determino para que en verdad sea un excelente ordenador, no solo depende del equipo sino de el software la estructura detallada de los códigos que entiende el computador.

El software sin lugar a dudas, es indispensable para las tareas a realizar, ya que de el depende el correcto funcionamiento de la computadora, y a pesar de los grandes avances que existen a la fecha, todavía falta mucho por prosperar. Pues el claro ejemplo se explico con Microsoft que con los sistemas operativos que ofrecen y los que existen en el mercado, todavía siguen evolucionando.

Tal como se expreso en el segundo capítulo los diferente tipos de lenguajes y códigos existentes, los que faltan por desarrollar aun, nos muestra el gran panorama de la programación aplicada en la automatización uno de los grandes destinos de cada programa, sin dejar a un lado las labores de un procesador de texto sin el cual no se podría haber tenido la facilidad de realizar esté trabajo recepcional, los reproductores multimedia que proporcionan un atractiva diversión, en fin la lista de los programas es de una gran extensión y no se podrían mencionar todos, o los que se pueden desarrollar para un fin especifico, todo depende del talento del programador.

Durante el tercer y cuarto capítulo se dio a conocer y la vida del video juego, el principal tema de esté trabajo. Se desarrollo una breve introducción a la historia del video juego, donde se trato de ver que también en esté tipo de

programas existe un gran evolución caminando a la par con su mercadotecnia y la gigantesca lucha entre los diseñadores y programadores de esté software.

Pero no todo fue la descripción de las empresas de video juegos sino que se fue desarrollando las ideas principales de un video juego y algunas de las alternativas existentes en la actualidad para crear un video juego.

A través de el progreso del video juego, los sonidos, sus gráficos, las visualizaciones que se logran en el, y los efectos que se pueden incluir, se examino Blitz 3D un programa de desarrollo por medio de un lenguaje de programación como alternativa para un video juego. Se explico de igual forma 3D Game Maker para diseño de video juegos, lo que lleva al desarrollo del proyecto del software de entretenimiento.

En el proyecto se realizo un breve tutorial de cómo se logra la composición del video juego **"Pacman Proy"**, y pasos muy importantes en la elaboración de una aspiración de esa magnitud, sus consecuencias al estar ejecutándolo, y la mentalidad que se debe tener en la realización de algo similar.

Esté proyecto llevo como meta concluir la ideas de que el video juego es tan fácil como jugarlo. Pues lleva una gran cantidad de tiempo, tener la mente bien enfocada en lo que se va ha trabajar, en fin quede la satisfacción en mi persona la realización del proyecto y el Trabajo Recepcional presente como muestra de mi esfuerzo y de la gran lucha y evolución por la que se lucha día tras día.

Axel Abraham Valdés Vargas.

GLOSARIO.

A.

Abrir: Orden de cargar un documento, copiándolo de un disco de memoria de la computadora.

Acceso aleatorio: Acceso inmediato a cualquier información en un dispositivo de de almacenamiento, sin importar su ubicación. También se le conoce como acceso directo.

Algoritmo: Conjunto de procedimientos paso a paso que, una vez completos resuelven un problema o llevan a cabo una tarea. Un programa de computadora generalmente comienza como algoritmo escrito en una versión limitada del inglés u otro lenguaje humano.

Almacenamiento primario: Memoria de la computadora.

Almacenamiento secundario: Tipo de almacenamiento en el que el ordenador registra información en forma semipermanente para que la PC u otra la pueda leerla después; incluye las unidades de discos de 3 ½ y CD´s.

Ambiente de programación: El espacio de trabajo de un programador; consta de un editor de textos, un compilador, un depurador, y otras utilerías de programación.

Aplicación o programa de aplicación: Herramienta de software, como un procesador de textos o una hoja de cálculo, que permite usar la computadora para un propósito especifico.

ASCII (American Standard Code for Information Interchange, pronúnciese "asqui"): El código más común para representar caracteres en la memoria de la PC. Representa cada carácter como un código único de 7 bits, más un octavo bit cuyo valor está determinado, por cuestiones técnicas, por los valores de los otros siete.

B.

Banal: Trivial, común, insustancial.

BASIC: (Beginner's All-purpose Symbolic Instruction Code) creado originalmente en la década de 1960 como lenguaje interactivo para aprender programación, Basic, es probablemente el lenguaje de programación de mayor difusión en la actualidad.

Binario: Sistema numérico de base 2 usado por las computadoras.

Bit: digito binario.

Byte: Colección de 8 bits. Un Byte puede representar 256 mensajes diferentes ($256= 2^8$). En muchas aplicaciones de cómputo, un Byte contiene un carácter de información.

C.

C: Lenguaje de programación que combina varías ventajas de lenguajes de alto nivel y lenguaje ensamblador. Su potencia, flexibilidad y eficacia lo han convertido en el lenguaje preferido entre los profesionales que programan en computadoras personales.

C++: Un popular dialecto del lenguaje de programación C con herramientas de programación orientada a objetos.

Cifrado: Proceso de codificación de datos para impedir el acceso no autorizado.

COBOL: (Common Business Oriented Language) Uno de los primeros lenguajes de alto nivel, diseñado específicamente para aplicaciones empresariales.

Codificación: Escritura de un programa a partir de algoritmos.

Código: Enunciado de programa que es el resultado de la codificación.

Compilador: Programa traductor que traduce todo un programa a lenguaje máquina antes de pasarlo a la computadora.

Computador personal (PC): Computadora de bajo costo del tamaño de una maquina de escribir, tan potente como muchas computadoras.

Computadora portátil: Computadora ligera, alimentada por baterías, diseñadas para la movilidad.

CPU: Unidad central de proceso, parte de la computadora que ejecuta el software.

Cualitativo: Que denota cualidad.

D.

Disco duro: Disco rígido, magnéticamente sensible, que gira a gran velocidad y en forma continúa dentro del gabinete de la computadora o en una caja conectado a la PC.

Disquete: Pequeña oblea de plástico flexible, magnéticamente sensible, empaquetada en un estuche de plástico. Es un medio común de acceso aleatorio para el almacenamiento de la computadora.

E.

Ensamblador: Programa que traduce instrucciones en lenguaje ensamblador a instrucciones en lenguaje máquina.

Epopeya: Conjunto de hechos gloriosos dignos de ser cantados épicamente.

Estación de trabajo: Computadora de alto nivel con poder computacional.

F.

Formato: Factores que determinan la apariencia de un documento o archivo.

G.

Guardar: Crear un archivo en disco y almacenarlo.

H.

Hardware: Las parte físicas de un sistema de cómputo.

HTML: Hyper Text Markup Lenguaje (lenguaje de marca de texto. Lenguaje para realizar páginas Web).

I.

Iconos: Imágenes que representan archivos, discos y otros elementos de las interfaz gráfica con el usuario.

Información: Formar, perfeccionar a alguien por medio de la instrucción y buena crianza.

Informática: Conjunto de conocimientos científicos y técnicas que hacen posible el tratamiento automático de la información por medio de ordenadores.

Interfaz gráfica con el usuario (GUI): Basada en imágenes gráficas y no en caracteres.

K.

Kilobyte: 1024 Bytes.

L.

Laptop: Computadora portátil.

LCD: pantallas planas de computadoras portátiles.

Lenguaje de alto nivel: lenguaje de programación más fácil de usar.

Lenguaje de bajo nivel: Lenguaje máquina u otro lenguaje que requiere un amplio conocimiento del mismo.

Lenguaje de cuarta generación: Lenguaje que no funciona con procedimientos; emplea frases y enunciados parecidos al inglés para emitir instrucciones.

Lenguaje máquina: Usado por los ordenadores para procesar instrucciones, formada por ceros y unos.

Lenguaje de programación: Conjunto de reglas que indican a la computadora que hacer.

Lenguaje ensamblador: Equivalente al lenguaje máquina, pero que los seres humanos pueden leer, escribir. En este lenguaje, los programadores

usan códigos alfabéticos que correspondan a las instrucciones numéricas de la máquina.

Lúdico: Perteneciente o relativo al juego.

M.

Máquina analítica: calculadora programable por Charles Babbage en el siglo XIX.

Mb: Megabyte 1000 Kb, un millón de Bytes.

Microprocesador: Circuito electrónico que actúa como unidad central de proceso de un ordenador, proporcionando el control de las operaciones de cálculo. Los microprocesadores también se utilizan en otros sistemas informáticos avanzados, como impresoras, automóviles o aviones.

Microsoft Windows: software de interfaz gráfica.

MS-DOS: Sistema operativo de disco de Microsoft con funcionamiento en base de comandos.

P.

Pascal: Lenguaje de alto nivel de amplia difusión de enseñanza.

Procesador de textos: En informática, aplicación utilizada para la manipulación de documentos basados en texto. Es el equivalente electrónico del papel, el bolígrafo, la máquina de escribir, el borrador y el diccionario.

Programa: Instrucciones de la computadora pasó a paso.

Programación: Escritura de programas.

Programación estructurada: Técnicas para facilitar la programación y reducir el número de errores; entre ellas está el uso de módulos y eliminación de enunciados.

Programación orientada a objetos: El programa es una colección de objetos que interactúan entre sí.

Programación visual: Creación de programas por medio de dibujos y apuntando a objetos en la pantalla.

Pseudocódigo: Lenguaje humano para analizar el principio de la programación. Cruza entre el lenguaje de computador y lenguaje natural que se escribe en algoritmos.

R.

RAM: Memoria de acceso aleatorio, se almacenan temporalmente instrucciones en la computadora.

ROM: Memoria de solo lectura.

S.

Scroll parallax: Es el efecto de realizar una panorámica o desplazamiento de la pantalla de juego con vista lateral. Cuando un juego posee varios planos de fondos móviles y funciona simulando, que los planos más lejanos se desplazan a menor velocidad.

Simulación: Uso de modelos por computadora para probar hipótesis y tomar decisiones.

Sintácticas: Perteneciente o relativo a la sintaxis.

Sistema operativo: Programas de administración de recursos para tener la operación eficiente del hardware.

Sistema operativo Macintosh: Primer sistema en utilizar la interfaz gráfica.

Sistematización: Organizar según un sistema.

Software: Instrucciones que indican a la computadora lo que debe hacer.

T.

Tajante: Concluyente, terminante, contundente.

Tópico: Perteneciente o relativo a determinado lugar.

U.

Ubicuidad: Que está presente a un mismo tiempo en todas partes.

Unidireccional: Que se orienta en una única dirección

Usuario: Dicho de una persona: Que tiene derecho de usar de una cosa ajena con cierta limitación.

V.

Variable: Porción nombrada de la memoria de la computadora cuyo contenido es examinado y modificado por el programa.

W.

World Wide Web: (www) Red de expansión mundial mejor conocida como Internet.

Windows: sistema operativo de interfaz grafica desarrollado por Microsoft.

BIBLIOGRAFIA.

LIBROS.

Maran, Graphics.
Computadoras Guía Visual.
5ta Edición.
ST. Editorial, Inc.
227 Pág.

Beekman, George.
COMPUTACIÓN & INFORMÁTICA HOY. Una mirada a la tecnología del mañana.
1995 ultima Edición.
ADDISON-WESLEY IBEROAMERICANA.
372 Pág.

Tiznado, Marco Antonio.
INFORMÁTICA.
Primera Edición.
Mc Graw-Hill Interamericana.
330 Pág.

Levine Gutiérrez, Guillermo.
COMPUTACIÓN Y PROGRAMACIÓN MODERNA. Perspectiva integral de la informática.
Segunda Edición.
Perarson educación.
640 Pág.

Deitel, Harvey M.
Deitel, Paul J.
C++ COMO PROGRAMAR.
Cuarta Edición.
Perarson educación.
1376 Pág.

Joyanes Aguilar, Luis.
FUNDAMENTOS DE PROGRAMACIÓN. Algoritmos, Estructura de datos y Objetos.
Primera Edición.
Mc Graw-Hill.
1003 Pág.

Rico Grau, Mercedes.
WINDOWS XP PASO A PASO.
Primera Edición.
Mc Graw-Hill.
119 Pág.

REVISTAS.

Medina, Marcos V.
VIDEOJUEGOS. Curso de Diseño y Programación.
N° 2.
01/Marzo/2003.
Editorial Iberprensa.
40 Pág.

EGM. Electronic Gaming Monthly en español.

Año 2, N° 10.
20/Octubre/2004.
Editorial Televisa, S.A de C.V.
95 Pág.

EGM. Electronic Gaming Monthly en español.
Año 3, N° 3.
22/Marzo/2005.
Editorial Televisa, S.A de C.V.
96 Pág.

Vive el egs. (Electronic Game Show) Master Player.
N° 31.
Diciembre 2004.
Mina editores y Corporativo mina S.A de C.V.
64 Pág.

SCIENTIFIC AMERICAN LATINOAMERICA.
Año 1, N° 1.
25/Abril/2002.
SCIENTIFIC AMERICAN LATINOAMERICA, S. DE R.L. DE C.V.
96 Pág.

QUO.
N° 28.
Febrero 2000.
Editorial Televisa.
118 Pág.

QUO.

N° 34.
Agosto 2000.
Editorial Televisa.
120 Pág.

MULTIMEDIA.

Biblioteca de Consulta Microsoft® Encarta® 2003.
© 1993-2002 Microsoft Corporation. Reservados todos los derechos.

www.speccy.org/spectrumforever/programas/aplicaciones/3d%20game%20maker/3dgamemaker.htm

http://game-maker.uptodown.com/

www.es.uu.hl/people/markov/gmaker/old/doc/gmaker_spanish_50.pdf

www.entidad-3d.com/tutorial_creación_3d/

www.gamemaker.nl

www.the3dgamemaker.com

www.config.freeuk.com/amiga/blitz/

www.blitzbasic.com/